UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/074,264	02/12/2002	Rajendra R. Damle	CET0006US	5023

33031 7590 01/14/2009

CAMPBELL STEPHENSON LLP
11401 CENTURY OAKS TERRACE
BLDG. H, SUITE 250
AUSTIN, TX 78758

EXAMINER	
PATEL, ASHOKKUMAR B	

ART UNIT	PAPER NUMBER
2456	

MAIL DATE	DELIVERY MODE
01/14/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/074,264
Filing Date: February 12, 2002
Appellant(s): DAMLE ET AL.

Mr. Jonathan N. Geld
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 10/20/2008 appealing from the Office action mailed 04/11/2008.

(1) Real Party in Interest

A statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5,680,400	York	10-1997
7,149,432	Smith	12-2006

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-11, 13-22 and 24-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over York (US 5, 680, 400) in view of Smith et al. (hereinafter Smith) (US 7, 149, 432 B1))

Referring to claim 1,

York teaches a method for transporting information over a network (Figs. 1-3) comprising:

decomposing an input datastream into a plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream (col. 1, line 50-57, "Data is presented to a data splitter. The data splitter separates the input data stream into N separate substreams by packaging data into packets, which may be of different sizes. As data is packetized, each packet is sent and presented to a separate data transmitter, one for each data substream, via an input queue to each transmitter. Each transmitter queue has a significant amount of packet storage available to hold input packets.")into one of a plurality of queues (Fig. 1, element TQ1, TQ2, TQN, col. 3, line

3-5, “Transmitters 112 and receivers 114 are connected to queues 110 and 116, each having the ability store multiple packets of data.”) and

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data (col. 3, line 30-38, “In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1.”),

forming each PDU by selecting the predetermined amount of data from the input datastream (col. 3, line 30-38, “ In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1.”), and

each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels (col. 3, line 39-48, “In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into

packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."); and

communicating said sub-streams between a first network element and a second network element of said network by transporting each one of said sub-streams over the corresponding channel (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. "), wherein

a transmission rate of said input datastream (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10 mega bytes per second.")is greater than a maximum transmission rate of any one of said channels (col. 4, line 19-23, "Transmitter data transfer rates typically range

between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

York teaches "decomposing an input datastream" and forming a data frame comprising one or more PDUs (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. and transmitting the data frame over the corresponding channel (As each substream is marked with its unique identity, all data packets in a given substream have the identity of the substream." As shown in Fig. 3).

York fails to teach "Smith decomposing an input datastream of a plurality of input datastreams , appending to each PDU a source identifier identifying the source of the input datastream, and said communicating comprises forming a data frame comprising one or more PDUs appended source identifier for each PDU and transmitting the data frame over the corresponding channel.

Smith decomposing an input datastream of a plurality of input datastreams (Fig. 2, col. 5, line 33-39, "Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit."), associating with each PDU

a source identifier identifying the source of the input datastream and said communicating comprises forming a data frame comprising one or more PDUs and the associated source identifiers for each PDU and transmitting the data frame over the corresponding channel ((Fig. 2, elements 14a-14c and 26a-26d, col. 5, line 33-39, "Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit." **Examiner notes that the source identifier for PDUs is nothing but "Q-IDs."** Which is illustrated on page 19 of 60/ 270, 444 as follows:

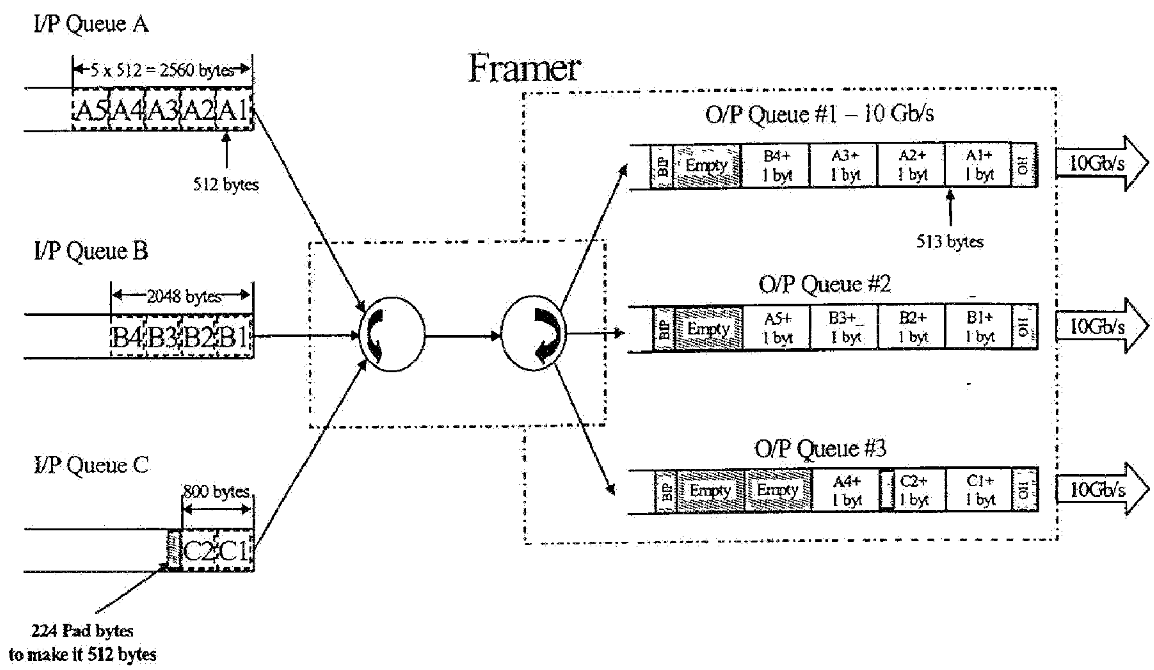


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

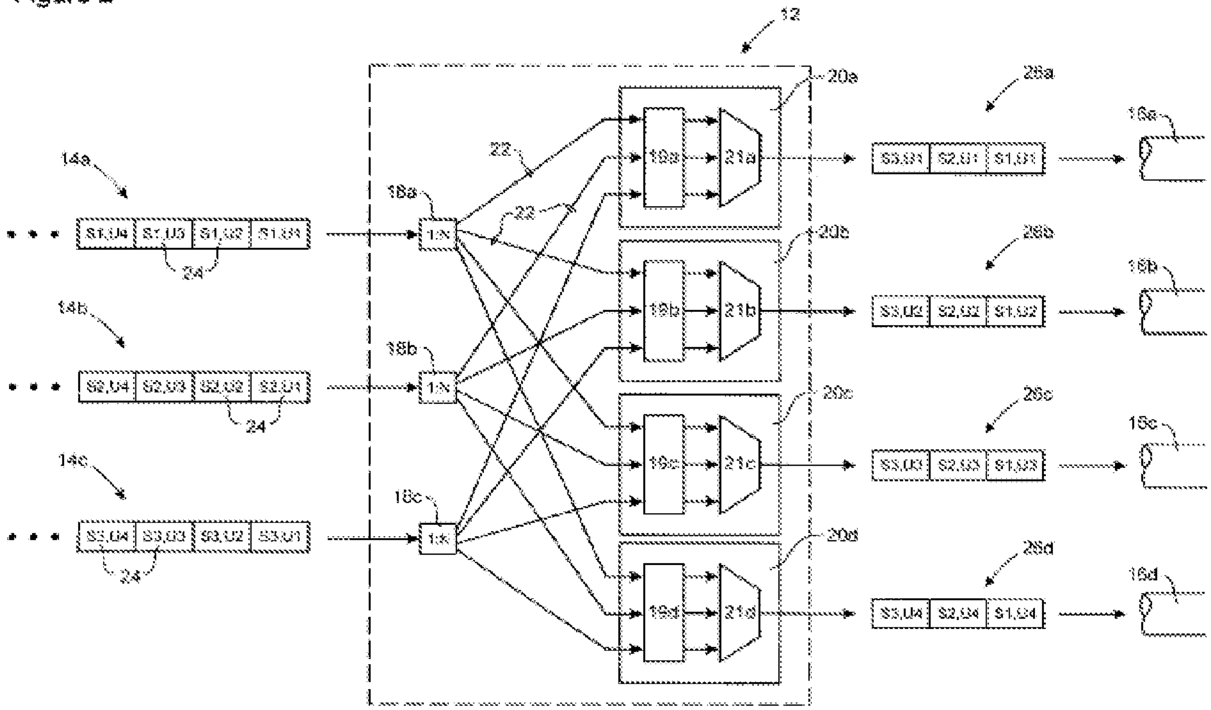
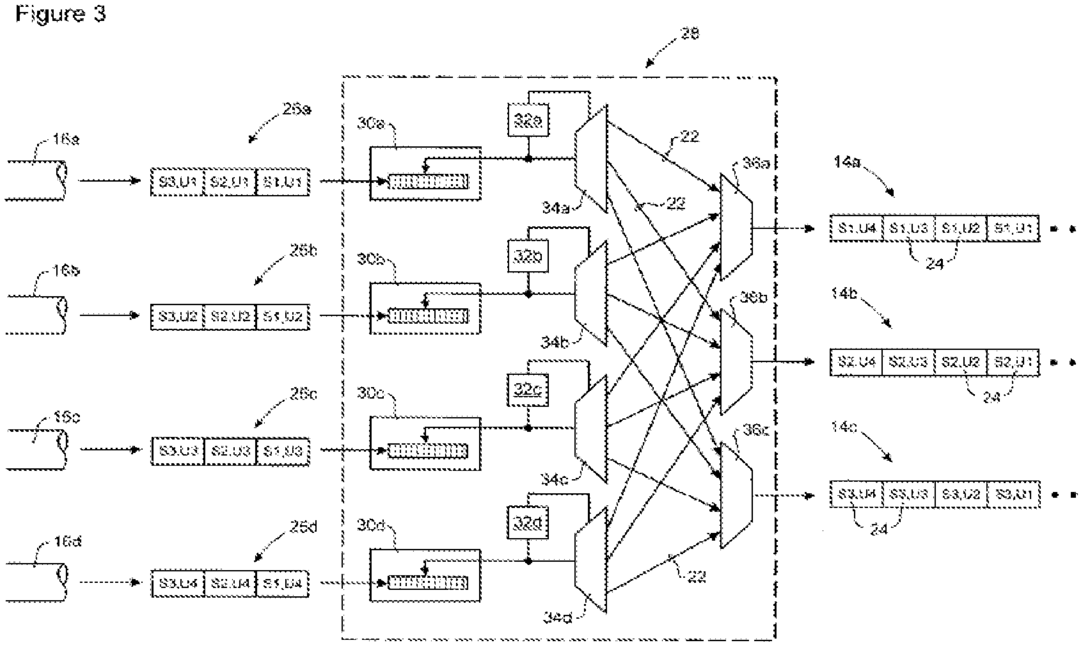


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, "The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. **The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,** and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of

<u>each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."</u>

<u>Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26</u>", as such identified by Smith as shown in Figs 2 and 3, S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated, <u>which is the source identifier for PDUs is nothing but "Q-IDs."</u>).

Because York and Smith teaches decomposing input datastreams, one incorporating the invention for "decomposing an input datastream" and the other incorporating the invention for "decomposing the plurality of input datastreams."

The difference of the invention lays into the inventive component they use, "Data Splitter" of York and "The signal distributor unit "of Smith. Therefore, it would have obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose.

Referring to claim 2,

York teaches the method of claim 1, wherein each of said channels is an optical channel (col. 3, line 61-67, "It is also assumed that standard transmission links 113 are employed such as data telephone lines, <u>fiber optic cables</u>, etc. Additionally, it is possible

to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.").

Referring to claim 3,

York teaches the method of claim 2, wherein each of said optical channels corresponds to a wavelength (col. 3, line 61-67, "It is also assumed that standard transmission links 113 are employed such as data telephone lines, <u>fiber optic cables</u>, etc. Additionally, it is possible to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.")

Referring to claim 4,

York teaches the method of claim 1, wherein said each one of said sub-streams has a transmission rate that is equal to or less than a maximum transmission rate of a corresponding one of said channels. (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10 mega bytes per second.", col. 4, line 19-23, "Transmitter data transfer rates typically range between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

Referring to claim 5,

York teaches the method of claim 1, further comprising: assembling said sub-streams into a reconstructed output datastream (col. 4, line 23-41, “Next, in steps 212, 214 receivers 114 receive packets from transmitter 112 via link 113. The transmission links are assumed to guarantee data packet ordering as presented by the input data packet queues. The receiving side initializes as many receivers as needed, or as many data receive substreams as are required, using as many receivers as are available. Each receiver guarantees that correct ordering of received packets in their respective queues 116. Receivers 114 also guarantee that data is integrity checked and are able to handle retransmitted packets as needed. Next, in step 216, the substream reassembly unit 118, polls each receiver queue 116 for data packets. Receiver queues 116 are polled in the same prearranged order as the round robin method described earlier. Then, the substream reassembly unit 118 reassembles the packets into a final output stream and is sent to FIFO 120 (typically the same size as FIFO 106) via bus 119. Then in step 218, the host processor 122 reads the continuous data stream.”)

Referring to claim 6,

York teaches the method of claim 5, wherein said assembling comprises: placing a portion of each of said substreams in a queue (Fig. 2, element 208), wherein said reconstructed output datastream is output by said queue (Fig. 2, element 214).

Referring to claim 7,

York teaches the method of claim 5, further comprising: performing protocol processing on said input datastream (col. 3, line 55-57, “It is assumed that each

transmitter are capable of sending packets of data to a receiver with data integrity checks and employ data compression if so configured."); and performing protocol processing on said reconstructed output datastream (Fig. 2, 210 and 212), wherein said protocol processing is performed using a protocol processor comprising a protocol stack (Fig. 1, elements 104 and 122).

Referring to claim 8,

York teaches the method of claim 1, further comprising: performing compression on a one of said sub-streams, wherein said one of said sub-streams has a transmission rate greater than a maximum transmission rate of the corresponding channel. (col. 3, line 55-57, "It is assumed that each transmitter are capable of sending packets of data to a receiver with data integrity checks and employ data compression if so configured.")

Referring to claim 9,

York teaches the method of claim 1, wherein said network is an existing network (Figs. 1 and 3).

Referring to claim 10,

York teaches the method of claim 1method of claim 1, wherein said network comprises an underlying network infrastructure (Fig. 1, element 113), and the method is performed without alteration of said underlying network infrastructure It is also assumed that standard transmission links 113 are employed such as data telephone lines, fiber optic cables, etc.", Col. 2, line 56-62, "The high performance communication system 100

takes a single input data stream 105, 107 from the host processor 104 and splits the single input data stream 105, 107 into multiple parallel streams 109 which are then presented to one or more independent physical data transmitters 112 and one or more independent physical data receivers 114.")

Referring to claim 11,

York teaches the method of claim 10, wherein said network comprises a fiber-optic system. (col. 3, line 61-67, "It is also assumed that standard transmission links 113 are employed such as data telephone lines, <u>fiber optic cables</u>, etc. Additionally, it is possible to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.").

Referring to claim 13,

York teaches a method for receiving information transported over a network (Fig. 1, element 118, Fig. 2, element 212) comprising:

receiving a plurality of sub-streams (Fig. 1, elements 116A……116N, 118, Fig. 2, element 212), wherein said sub-streams are created by decomposing an input datastream into a plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream (col. 1, line 50-57, " Data is presented to a data splitter. The data splitter separates the input data stream into N separate substreams by packaging data into packets, which may be of different sizes. As data is packetized, each packet is sent and presented to a separate data transmitter, one for each data

substream, via an input queue to each transmitter. Each transmitter queue has a significant amount of packet storage available to hold input packets.") into one of a plurality of queues (Fig. 1, element TQ1, TQ2,.......... TQN, col. 3, line 3-5, "Transmitters 112 and receivers 114 are connected to queues 110 and 116, each having the ability store multiple packets of data.") and

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data (col. 3, line 30-38, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."),

forming each PDU by selecting the predetermined amount of data from the input datastream (col. 3, line 30-38, " In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and <u>is determined at the time the network connections are established.</u> FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."), and

each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."); and

each of said sub-streams is transported over said network on the corresponding channel (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. "), and

a transmission rate of said input datastream (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10

mega bytes per second.")is greater than a maximum transmission rate of any one of said channels (col. 4, line 19-23, "Transmitter data transfer rates typically range between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

assembling said sub-streams into a reconstructed output datastream (col. 4, line 34-41, "Next, in step 216, the substream reassembly unit 118, polls each receiver queue 116 for data packets. Receiver queues 116 are polled in the same prearranged order as the round robin method described earlier. Then, the substream reassembly unit 118 reassembles the packets into a final output stream and is sent to FIFO 120 (typically the same size as FIFO 106) via bus 119. Then in step 218, the host processor 122 reads the continuous data stream.")

York teaches "decomposing an input datastream" and forming a data frame comprising one or more PDUs (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. and transmitting the data frame over the corresponding channel (As each substream is marked with its unique identity, all data packets in a given substream have the identity of the substream." As shown in Fig. 3).

York fails to teach "Smith decomposing an input datastream <u>of a plurality of input datastreams </u>, appending to each PDU a source identifier identifying the source of the input datastream, and said communicating comprises forming a data frame

comprising one or more PDUs and the appended source identifier for each PDU and transmitting the data frame over the corresponding channel.

Smith decomposing an input datastream of a plurality of input datastreams (Fig. 2, col. 5, line 33-39, “Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit.”), associating with each PDU a source identifier identifying the source of the input datastream and said communicating comprises forming a data frame comprising one or more PDUs and the associated source identifiers for each PDU and transmitting the data frame over the corresponding channel ((Fig. 2, elements 14a-14c and 26a-26d, col. 5, line 33-39, “Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit.” **Examiner notes that the source identifier for PDUs is nothing but “Q-IDs.”** Which is illustrated on page 19 of 60/ 270, 444 as follows:

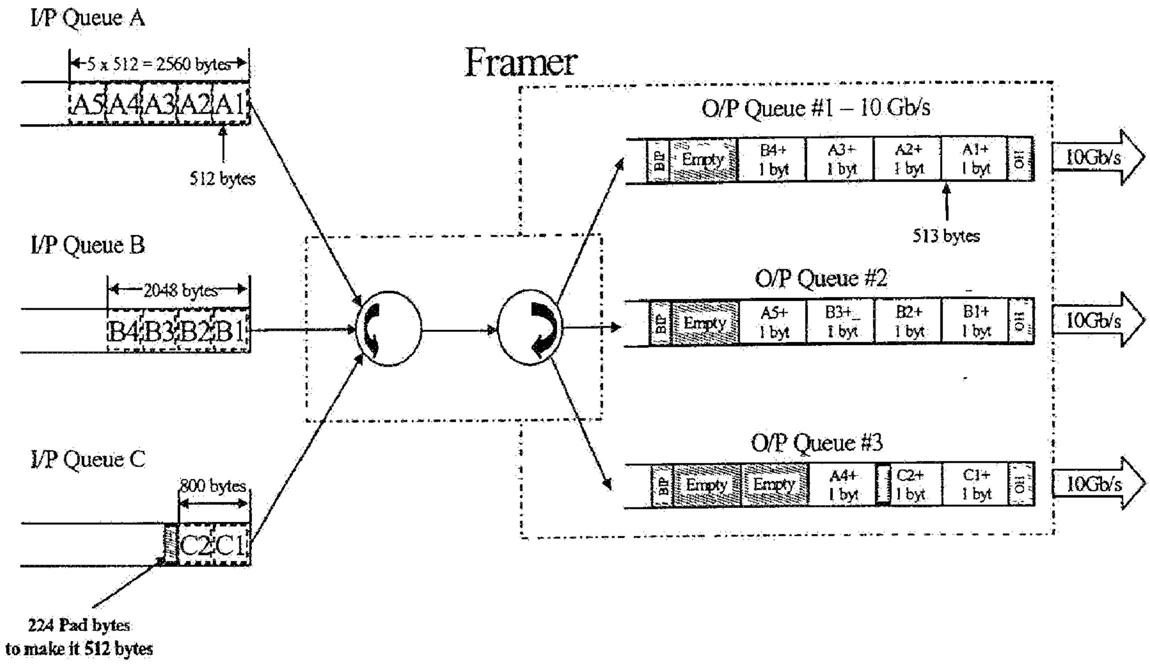


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

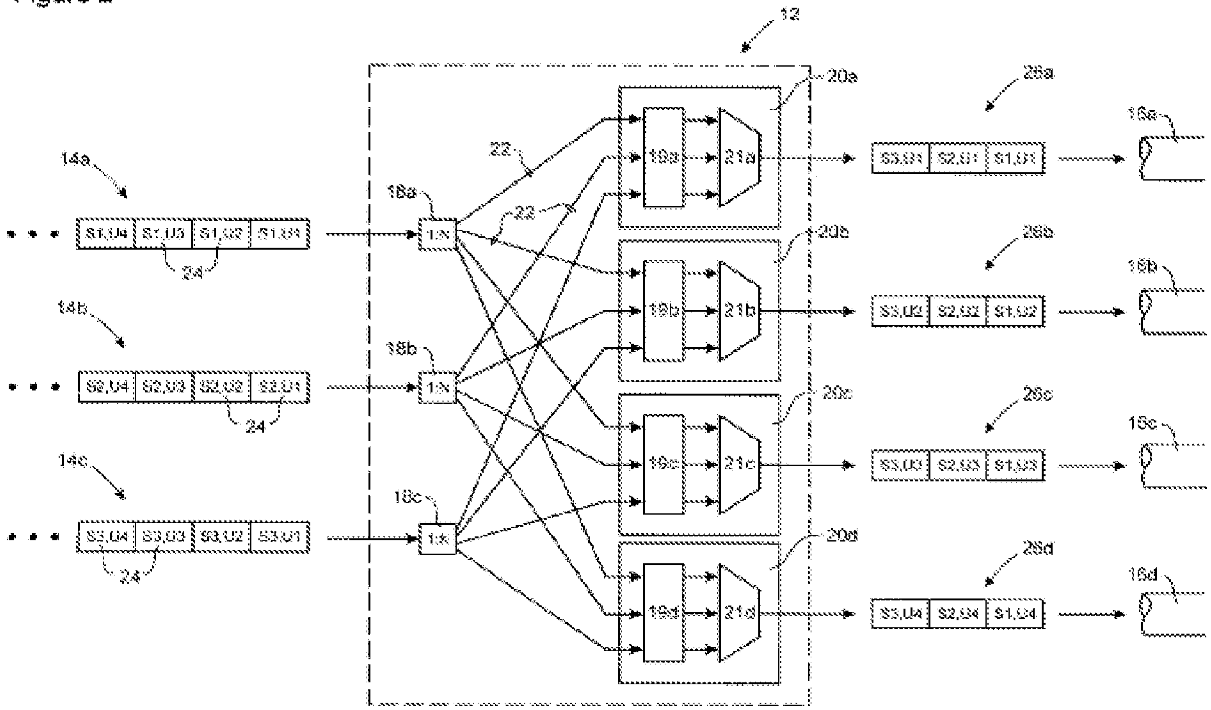
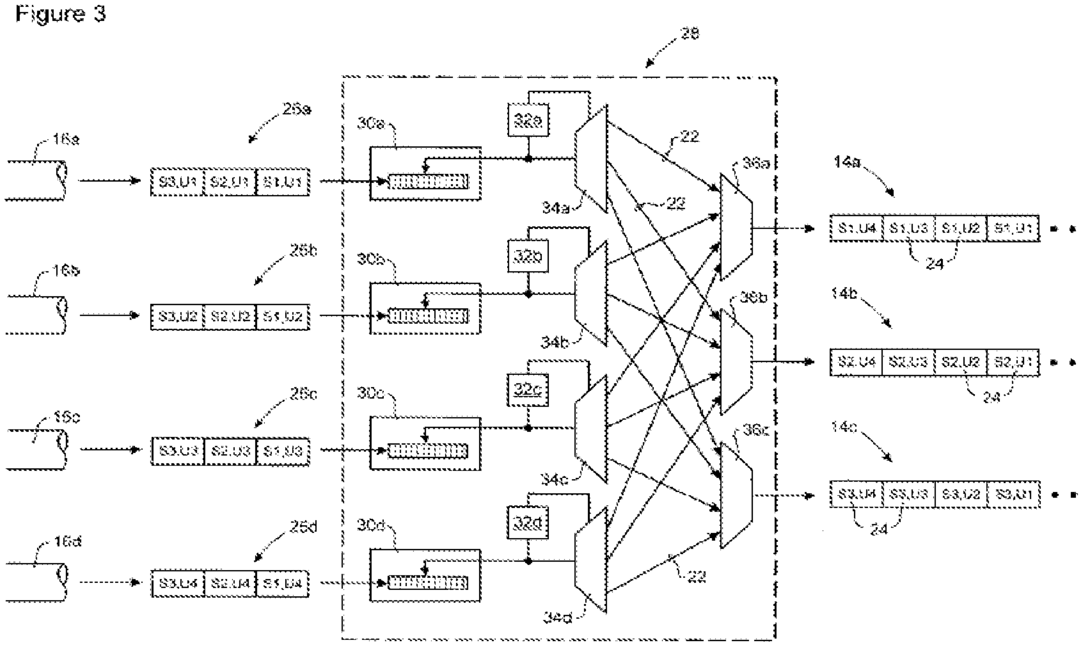


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, "The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. <u>**The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,** and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of</u>

each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."

Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated , which is the source identifier for PDUs is nothing but "Q-IDs.").

Because York and Smith teaches decomposing input datastreams, one incorporating the invention for " decomposing an input datastream" and the other incorporating the invention for "decomposing the plurality of input datastreams."

The difference of the invention lays into the inventive component they use, "Data Splitter" of York and "The signal distributor unit " of Smith. Therefore, it would have obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose.

Referring to claim 14,

York teaches the method of claim 13, wherein each of said channels is an optical channel (col. 3, line 61-67, "It is also assumed that standard transmission links 113 are employed such as data telephone lines, fiber optic cables, etc. Additionally, it is possible

to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.").

Referring to claim 15,

York teaches the method of claim 14, wherein each of said optical channels corresponds to a wavelength (col. 3, line 61-67, "It is also assumed that standard transmission links 113 are employed such as data telephone lines, fiber optic cables, etc. Additionally, it is possible to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.")

Referring to claim 16,

York teaches the method of claim 13, wherein said each one of said sub-streams has a transmission rate that is equal to or less than a maximum transmission rate of a corresponding one of said channels. (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10 mega bytes per second.", col. 4, line 19-23, "Transmitter data transfer rates typically range between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

Referring to claim 17,

York teaches the method of claim 5, wherein said assembling comprises: placing a portion of each of said substreams in a queue (Fig. 2, element 208), wherein said reconstructed output datastream is output by said queue (Fig. 2, element 214).

Referring to claim 18,

York teaches the method of claim 13, further comprising: decomposing said input datastream into said sub-streams; and transporting said each of said sub-streams over said network on the corresponding channel. (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. ")

Referring to claim 19,

York teaches the method of claim 13, further comprising: performing protocol processing on said input datastream (col. 3, line 55-57, "It is assumed that each transmitter are capable of sending packets of data to a receiver with data integrity

checks and employ data compression if so configured."); and performing protocol processing on said reconstructed output datastream (Fig. 2, 210 and 212), wherein said protocol processing is performed using a protocol processor comprising a protocol stack (Fig. 1, elements 104 and 122).

Referring to claim 20,

York teaches the method of claim 13, wherein said network is an existing network (Figs. 1 and 3).

Referring to claim 21,

York teaches the method of claim 13, wherein said network comprises an underlying network infrastructure (Fig. 1, element 113), and the method is performed without alteration of said underlying network infrastructure It is also assumed that standard transmission links 113 are employed such as data telephone lines, fiber optic cables, etc.", Col. 2, line 56-62, "The high performance communication system 100 takes a single input data stream 105, 107 from the host processor 104 and splits the single input data stream 105, 107 into multiple parallel streams 109 which are then presented to one or more independent physical data transmitters 112 and one or more independent physical data receivers 114.")

Referring to claim 22,

York teaches the method of claim 21, wherein said network comprises a fiber-optic system. (col. 3, line 61-67, "It is also assumed that standard transmission links 113

are employed such as data telephone lines, fiber optic cables, etc. Additionally, it is possible to have one link 113 or multiple links corresponding to the quantity of transmitters and receivers. If a single link is employed, then it is necessary to mark each substream packet with a unique identity and multiplex the transmitters.").

Referring to claim 24,

York teaches an apparatus for transporting information over a network comprising:

a first sub-stream management device (Fig. 1, elements 102, 103, 106, 108 and 110A....110N), comprising an input configured to receive an input datastream (Fig. 1, element 104), and a plurality of outputs (Fig. 1, element 108, 11- A.......110N), wherein each of said outputs is c6nfigur~d to output one of a plurality of sub-streams (Fig.1, elements 110A.......100n, 112A...........112N), wherein the input datastream is decomposed to form the plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream (col. 1, line 50-57, " Data is presented to a data splitter. The data splitter separates the input data stream into N separate substreams by packaging data into packets, which may be of different sizes. As data is packetized, each packet is sent and presented to a separate data transmitter, one for each data substream, via an input queue to each transmitter. Each transmitter queue has a significant amount of packet storage available to hold input packets.")into one of a plurality of queues (Fig. 1, element TQ1, TQ2,.......... TQN, col. 3, line 3-5,

"Transmitters 112 and receivers 114 are connected to queues 110 and 116, each having the ability store multiple packets of data.") and

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data (col. 3, line 30-38, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."),

forming each PDU by selecting the predetermined amount of data from the input datastream (col. 3, line 30-38, " In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and <u>is determined at the time the network connections are established.</u> FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."), and

each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into

packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."); and

each of said sub-streams is transported over said network on the corresponding channel (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. "), and

a transmission rate of said input datastream (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10 mega bytes per second.")is greater than a maximum transmission rate of any one of said channels (col. 4, line 19-23, "Transmitter data transfer rates typically range

between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

York teaches "decomposing an input datastream" and forming a data frame comprising one or more PDUs (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. and transmitting the data frame over the corresponding channel (As each substream is marked with its unique identity, all data packets in a given substream have the identity of the substream." As shown in Fig. 3).

York fails to teach "Smith decomposing an input datastream <u>of a plurality of input datastreams </u>, appending to each PDU a source identifier identifying the source of the input datastream, and said communicating comprises forming a data frame comprising one or more PDUs and <u>the appended source identifier for each PDU</u> and transmitting the data frame over the corresponding channel.

Smith decomposing an input datastream of a plurality of input datastreams (Fig. 2, col. 5, line 33-39, "Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit."), associating with each PDU

a source identifier identifying the source of the input datastream and said communicating comprises forming a data frame comprising one or more PDUs and the associated source identifiers for each PDU and transmitting the data frame over the corresponding channel ((Fig. 2, elements 14a-14c and 26a-26d, col. 5, line 33-39, “Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit.” **Examiner notes that the source identifier for PDUs is nothing but “Q-IDs.”** Which is illustrated on page 19 of 60/ 270, 444 as follows:

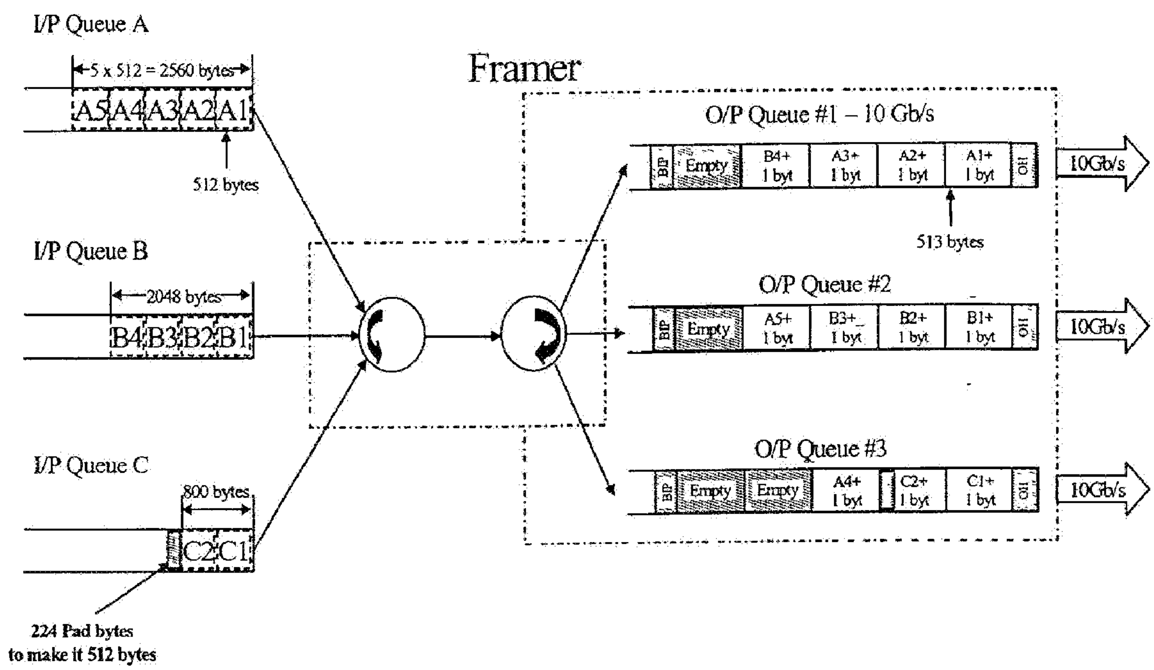


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

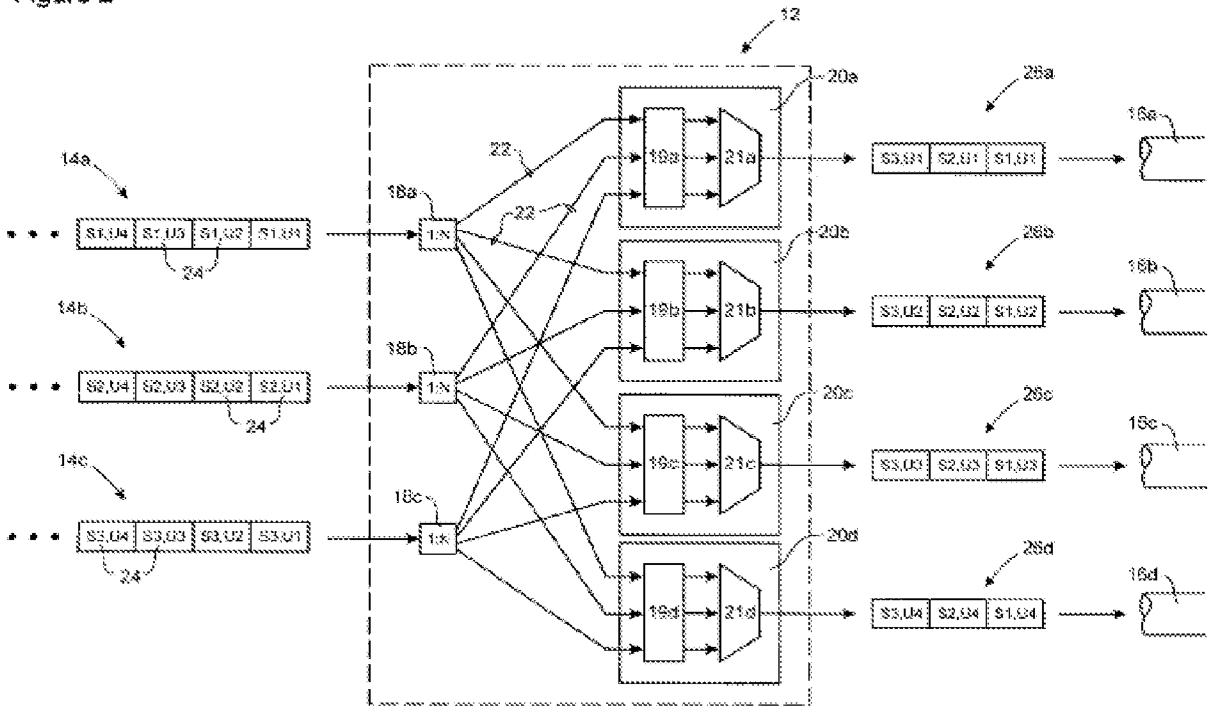
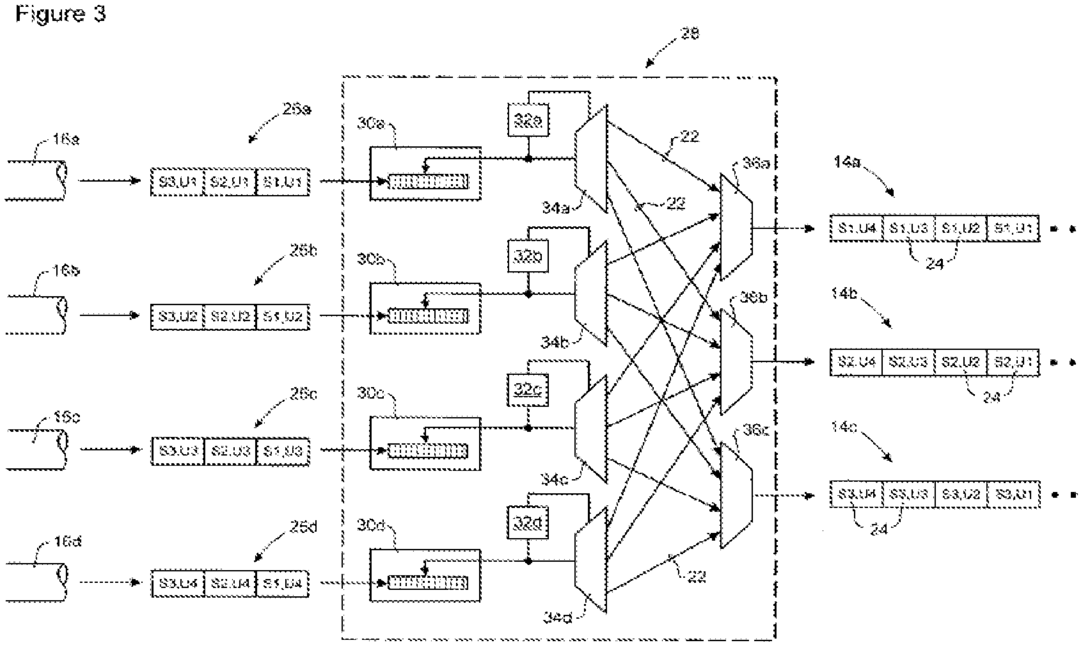


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, "The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. <u>**The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,** and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of</u>

each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."

Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated , which is the source identifier for PDUs is nothing but "Q-IDs.").

Because York and Smith teaches decomposing input datastreams, one incorporating the invention for " decomposing an input datastream" and the other incorporating the invention for "decomposing the plurality of input datastreams."

The difference of the invention lays into the inventive component they use, "Data Splitter" of York and "The signal distributor unit " of Smith. Therefore, it would have obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose.

Referring to claim 25,

Claim 25 is a claim to an apparatus for carrying t out the method of claim 14. Therefore claim 25 is rejected for the reasons set forth for claim 14.

Referring to claim 26,

Claim 26 is a claim to an apparatus for carrying t out the method of claim 15. Therefore claim 26 is rejected for the reasons set forth for claim 15.

Referring to claim 27,

Claim 27 is a claim to an apparatus for carrying t out the method of claim 16. Therefore claim 27 is rejected for the reasons set forth for claim 16.

Referring to claim 28,

York teaches the apparatus of claim 24, further comprising a second sub-stream management device (Fig. 1, elements, 114A…..114N, 116A….116N, element 118, element 120), comprising an output configured to output a reconstructed output datastream (Fig. 1, element 118), and a plurality of inputs(Fig. 1, elements, 114A…..114N, 116A….116), wherein each of said inputs is configured to receive one of said sub- streams (Fig. 1, elements, 114A…..114N, 116A….116N); and an underlying network infrastructure, communicatively coupled to said first and said second sub-stream management devices, and comprising said channels (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as will be used during the data transmission phase of data transfer.

This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. “).

Referring to claim 29,

York teaches the apparatus of claim 28, further comprising

a first protocol processor, coupled to said input (col. 3, line 55-57, “It is assumed that each transmitter are capable of sending packets of data to a receiver with data integrity checks and employ data compression if so configured.” (Fig. 2, 210 and 212); a second protocol processor, coupled to said output; and wherein, the first and second protocol processors each comprise a protocol stack. (Fig. 1, elements 104 and 122 both including 108 and 118).

Referring to claim 30,

York teaches an apparatus for transporting information over a network comprising:

a first sub-stream management device(Fig. 1, elements 114A…114N, 116A…..116N, 118, 120 and 121), comprising an output configured to output a reconstructed output datastream(Fig. 1, elements 114A…114N, 116A…..116N, 118, 120 and 121), and a plurality of inputs(Fig. 1, elements 114A…114N, 116A…..116N, 118, 120 and 121), wherein each of said inputs is configured to receive one of a plurality

of sub-streams (Fig. 1, elements 114A…114N, 116A…..116N, 118, 120 and 121),wherein

said decomposing comprises placing a portion of the input datastream (col. 1, line 50-57, " Data is presented to a data splitter. The data splitter separates the input data stream into N separate substreams by packaging data into packets, which may be of different sizes. As data is packetized, each packet is sent and presented to a separate data transmitter, one for each data substream, via an input queue to each transmitter. Each transmitter queue has a significant amount of packet storage available to hold input packets.")into one of a plurality of queues (Fig. 1, element TQ1, TQ2,………. TQN, col. 3, line 3-5, "Transmitters 112 and receivers 114 are connected to queues 110 and 116, each having the ability store multiple packets of data.") and

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data (col. 3, line 30-38, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."),

forming each PDU by selecting the predetermined amount of data from the input datastream (col. 3, line 30-38, " In step 204, the data splitter 108 splits the single

continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and <u>is determined at the time the network connections are established.</u> FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."), and

each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. The packet size is dependent on the characteristics of the transmitters and receivers and is determined at the time the network connections are established. FIG. 3 shows part of a continuous data stream and an example packet. In this description and throughout the figures, N represents any number greater than 1."); and

each of said sub-streams is transported over said network on the corresponding channel (col. 5, line 4-17, "During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used

and the ordering of logical channels as will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. "), and

a transmission rate of said input datastream (col. 3, line 24-26, "To transfer a file, the host processor 104 sends a continuous data stream of data, at rates around 1-to-10 mega bytes per second.")is greater than a maximum transmission rate of any one of said channels (col. 4, line 19-23, "Transmitter data transfer rates typically range between 1.54 mega bits per second to 10 mega bits per second. Higher throughput can be achieved by increasing the number of transmitters 112 and receivers 114.")

York teaches "decomposing an input datastream" and forming a data frame comprising one or more PDUs (col. 3, line 39-48, "In step 204, the data splitter 108 splits the single continuous data stream into N separate substreams by packaging the data into packets, which may be variable sizes. and transmitting the data frame over the corresponding channel (As each substream is marked with its unique identity, all data packets in a given substream have the identity of the substream." As shown in Fig. 3).

York fails to teach "Smith decomposing an input datastream <u>of a plurality of input datastreams</u> , appending to each PDU a source identifier identifying the source of the input datastream, and said communicating comprises forming a data frame comprising one or more PDUs and <u>the appended source identifier for each</u>

PDU and transmitting the data frame over the corresponding channel.

Smith decomposing an input datastream of a plurality of input datastreams (Fig. 2, col. 5, line 33-39, “Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit.”), associating with each PDU a source identifier identifying the source of the input datastream and said communicating comprises forming a data frame comprising one or more PDUs and the associated source identifiers for each PDU and transmitting the data frame over the corresponding channel ((Fig. 2, elements 14a-14c and 26a-26d, col. 5, line 33-39, “Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit.” **Examiner notes that the source identifier for PDUs is nothing but “Q-IDs.”** Which is illustrated on page 19 of 60/ 270, 444 as follows:

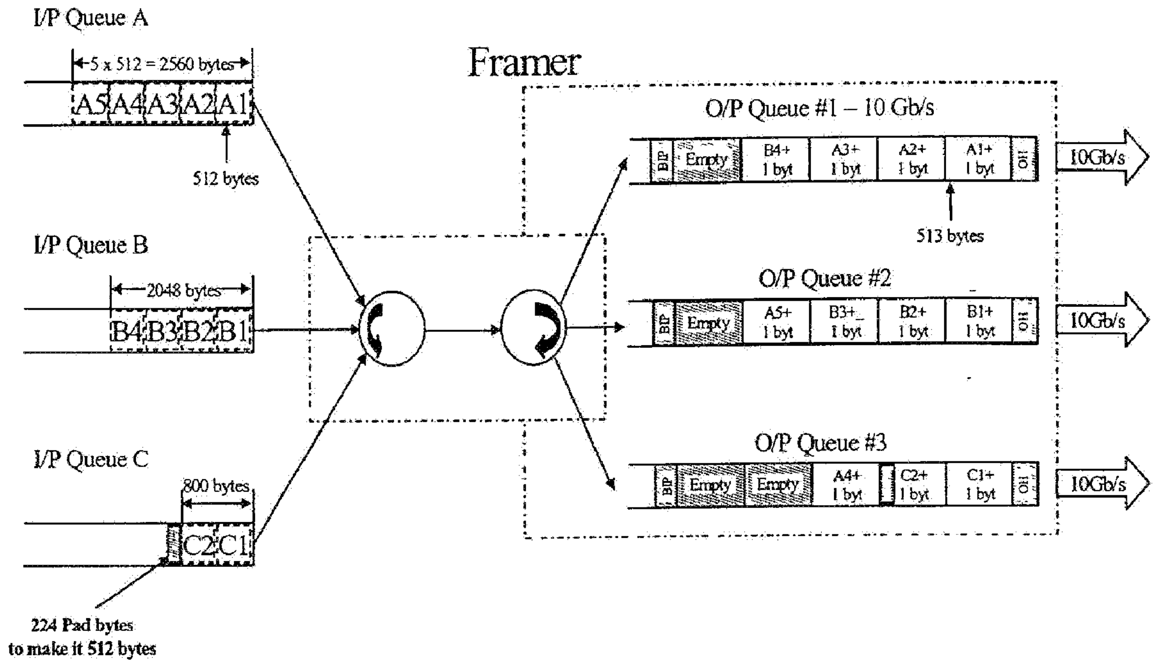


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

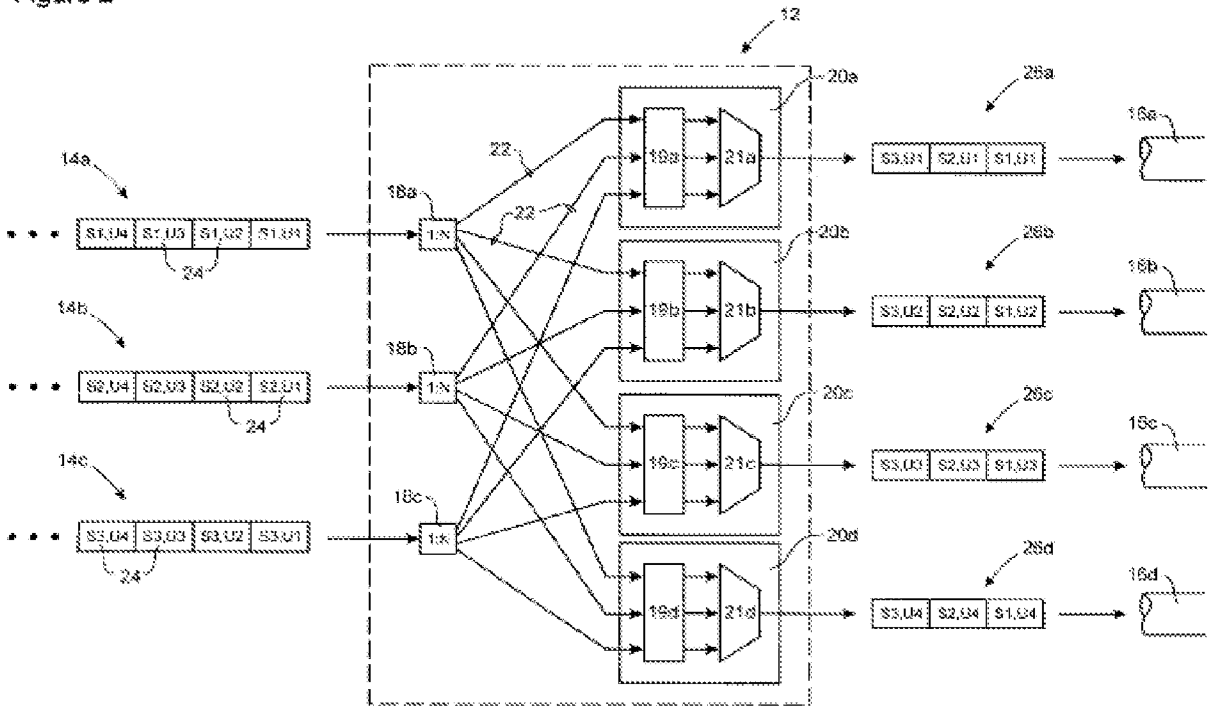
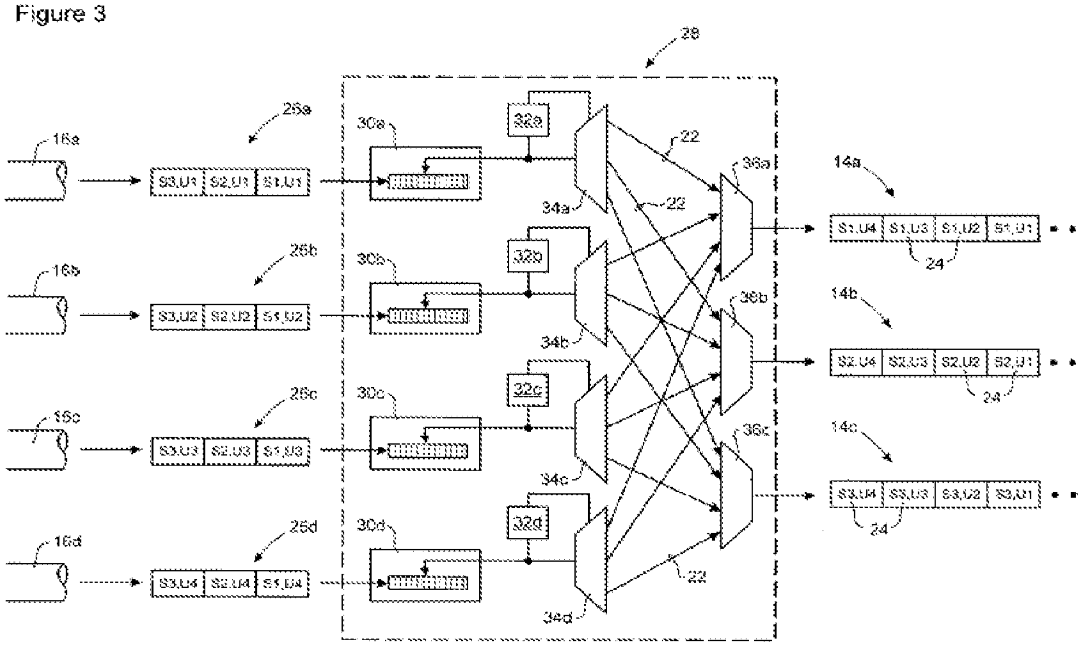


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, “The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. **The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,** and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of

<u>each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."</u>

<u>**Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26**</u>**", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated ,** <u>**which is the source identifier for PDUs is nothing but "Q-IDs."**</u>).

Because York and Smith teaches decomposing input datastreams, one incorporating the invention for " decomposing an input datastream" and the other incorporating the invention for "decomposing the plurality of input datastreams."

The difference of the invention lays into the inventive component they use, "Data Splitter" of York and "The signal distributor unit " of Smith. Therefore, it would have obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose.

Referring to claim 31,

Claim 31 is a claim to an apparatus for carrying t out the method of claim 14. Therefore claim 31 is rejected for the reasons set forth for claim 14.

Referring to claim 32,

Claim 32 is a claim to an apparatus for carrying t out the method of claim 15. Therefore claim 32 is rejected for the reasons set forth for claim 15.

Referring to claim 33,

Claim 33 is a claim to an apparatus for carrying t out the method of claim 16. Therefore claim 33 is rejected for the reasons set forth for claim 16.

Referring to claim 34,

York teaches the apparatus of claim 30, further comprising a second sub-stream management device (Fig. 1, elements, 102, 104, 106, 108, 110A…..110N, 112A…..112N), comprising an input configured to receive said input datastream(Fig. 1, elements, 102, 104, 106, 108, 110A…..110N, 112A…..112N), and a plurality of outputs(Fig. 1, elements, 102, 104, 106, 108, 110A…..110N, 112A…..112N), wherein each of said outputs is configured to output one of said sub-streams (Fig. 1, elements, 102, 104, 106, 108, 110A…..110N, 112A…..112N),; and an underlying network infrastructure, communicatively coupled to said first and said second sub-stream management devices, and comprising said channels(col. 5, line 4-17, “During network connection setup, the dam splitter interrogates each logical path to determine the allowable maximum data packet size permitted by the underlying physical transport system. By design convention, the data splitter selects one of the logical connections as the first connection to the other host computer. This connection is established first and involves the sending of greetings and setup parameters between the data splitter and the data packet reassembly unit. Setup parameters include the number and exact identity of logical channels which will be used and the ordering of logical channels as

will be used during the data transmission phase of data transfer. This is important as the data reassembly unit must have the identical number of logical channels as the transmission side and it must use them in the identical order to the transmit side. ").

Referring to claim 35,

York teaches the apparatus of claim 34, further comprising

a first protocol processor, coupled to said input (col. 3, line 55-57, "It is assumed that each transmitter are capable of sending packets of data to a receiver with data integrity checks and employ data compression if so configured." (Fig. 2, 210 and 212); a second protocol processor, coupled to said output; and wherein, the first and second protocol processors each comprise a protocol stack. (Fig. 1, elements 104 and 122 both including 108 and 118).

Referring to claim 36,

York teaches the method of claim 1 wherein selecting the selected one of a plurality of channels comprises: using a simple round-robin technique to choose an available one of the plurality of channels (col. 1, line 58-col. 2, line3, "Data is sent to the array of transmitters in round-robin fashion such that the data is first presented to the first transmitter, then to the second transmitter, and so on until each transmitter has been sent a packet, then the first transmitter is sent another, and so on, until all data packets have been sent to a transmitter. Each data transmitter processes packets and transmits them sequentially to one or more data receivers. There may be more than one physical media between the transmitters and receivers, or a single transmission link

may be used with all data substreams being multiplexed together. As each substream is marked with its unique identity, all data packets in a given substream have the identity of the substream.")

Referring to claim 37,

York teaches the apparatus of Claim 24 wherein selecting the selected one of the plurality of outputs comprises: using a simple round-robin technique to choose an available one of the plurality of outputs (col. 4, line 34-41, "Next, in step 216, the substream reassembly unit 118, polls each receiver queue 116 for data packets. Receiver queues 116 are polled in the same prearranged order as the round robin method described earlier. Then, the substream reassembly unit 118 reassembles the packets into a final output stream and is sent to FIFO 120 (typically the same size as FIFO 106) via bus 119. Then in step 218, the host processor 122 reads the continuous data stream.")

Claim Rejections - 35 USC § 102

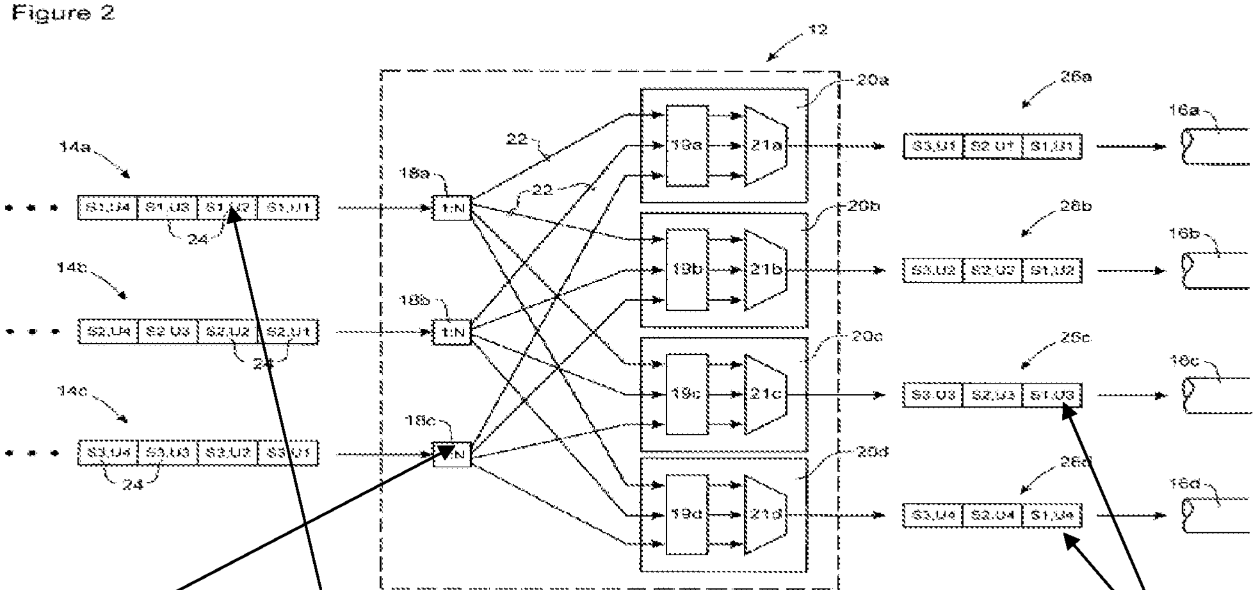
3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:
A person shall be entitled to a patent unless-

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1, 13, 24 and 30 are rejected under 35 U.S.C. 102(e) as being anticipated by Smith et al. (hereinafter Smith) (US 7, 149, 432 B1).

Referring to claim 1, Smith teaches a method for transporting information over a network comprising:

Figure 2



decomposing an input datastream of a plurality of input datastreams into a plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues, forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data, forming each PDU by selecting the predetermined amount of data from the input datastream, and appending to each PDU a source identifier identifying the source of the input datastream and each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels; and communicating said sub-streams between a first network element and a second network element of said network by transporting each one of said sub- streams over the corresponding channel, wherein a transmission rate of said input datastream is greater than a maximum transmission rate of any one of said channels, and said communicating comprises forming a data frame comprising one or more PDUs and the appended source identifiers for each PDU and transmitting the data frame over the corresponding channel. (Col. 5, line 5-col. 7, line 55, col. 5, line 18-27, “The channels 16 can then be routed through conventional signal transmission circuits (not shown) which perform electro/optical conversion and optical multiplexing of the channels 16 into the fiber links 6a e in a manner known in the art. For example, known signal multiplexing methods may be utilized to multiplex the channels 16 onto respective different wavelengths within the same WDM or DWDM fiber, respective wavelengths within two or more WDM or DWDM fibers, or respective single-wavelength fibers.”)

<u>Examiner notes that the source identifier for PDUs is nothing but "Q-IDs."</u> Which is illustrated on page 19 of 60/ 270, 444 as follows:

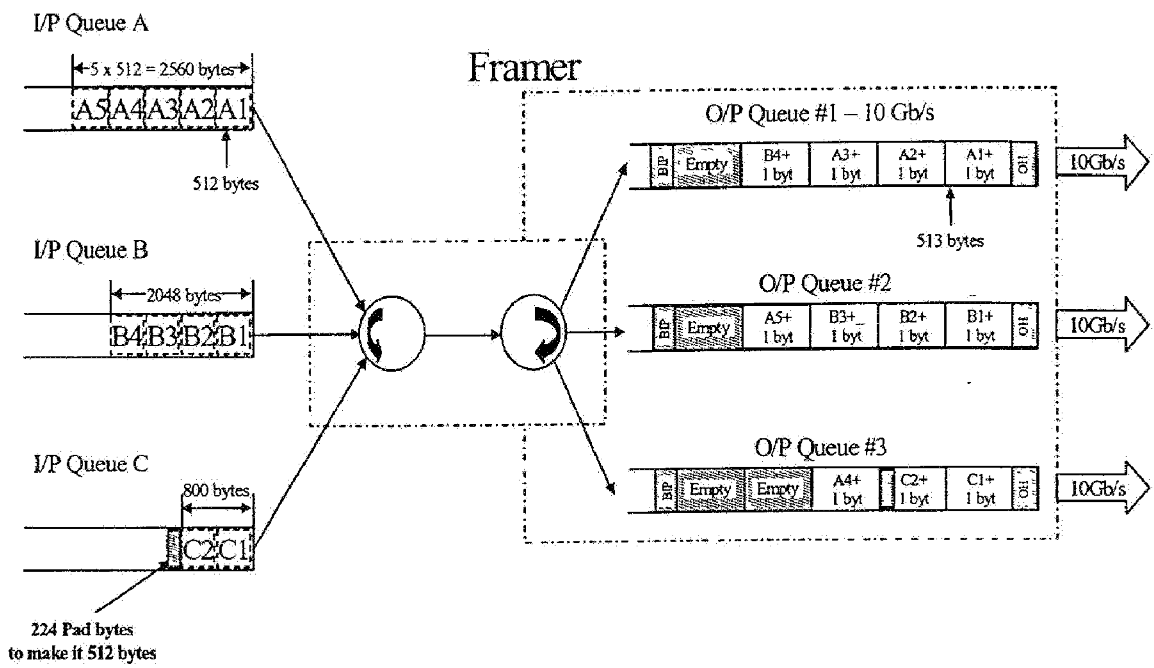


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

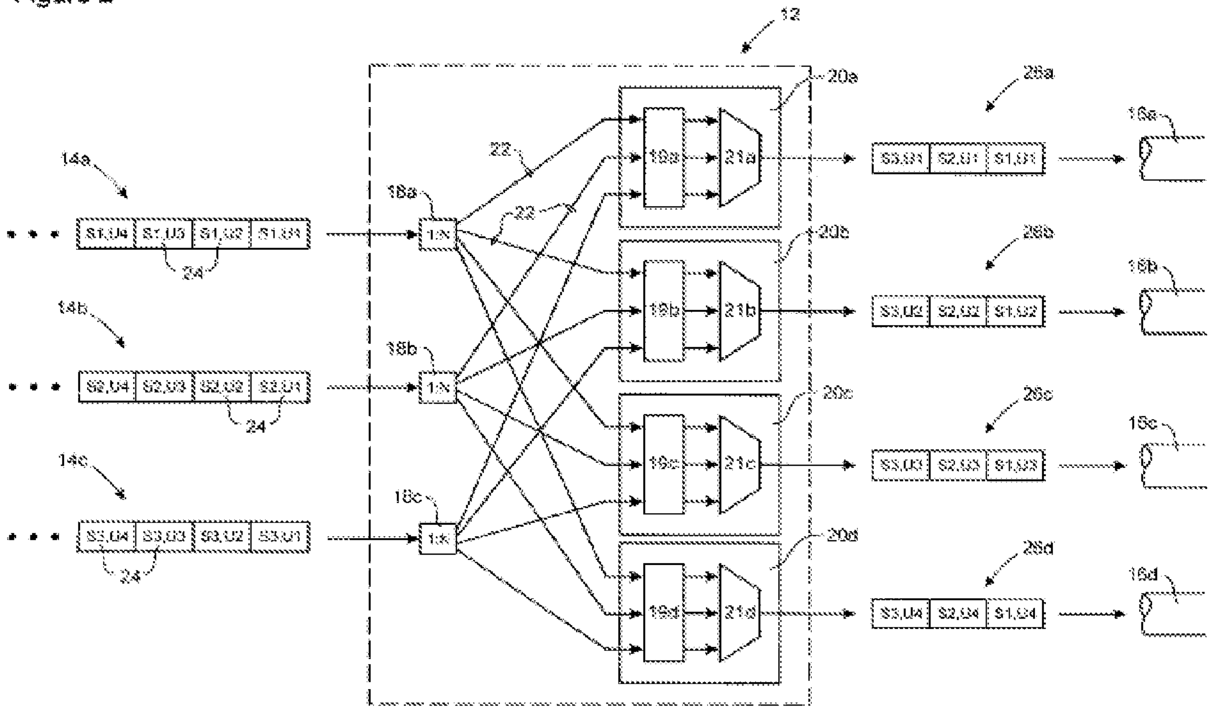
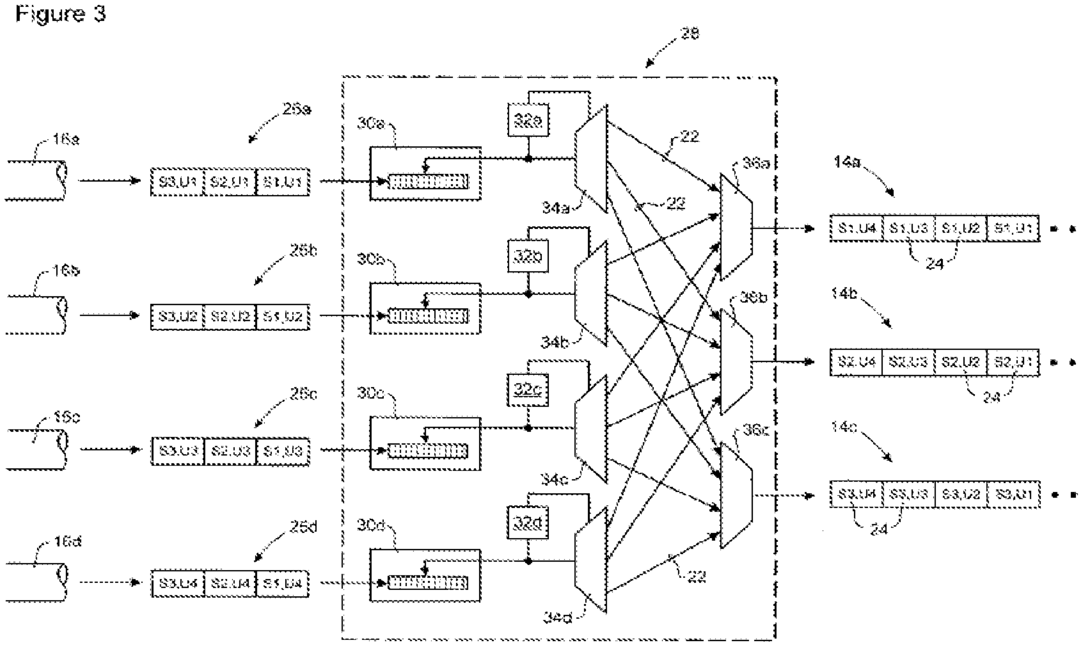


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, "The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. **<u>The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,</u>** <u>and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of</u>

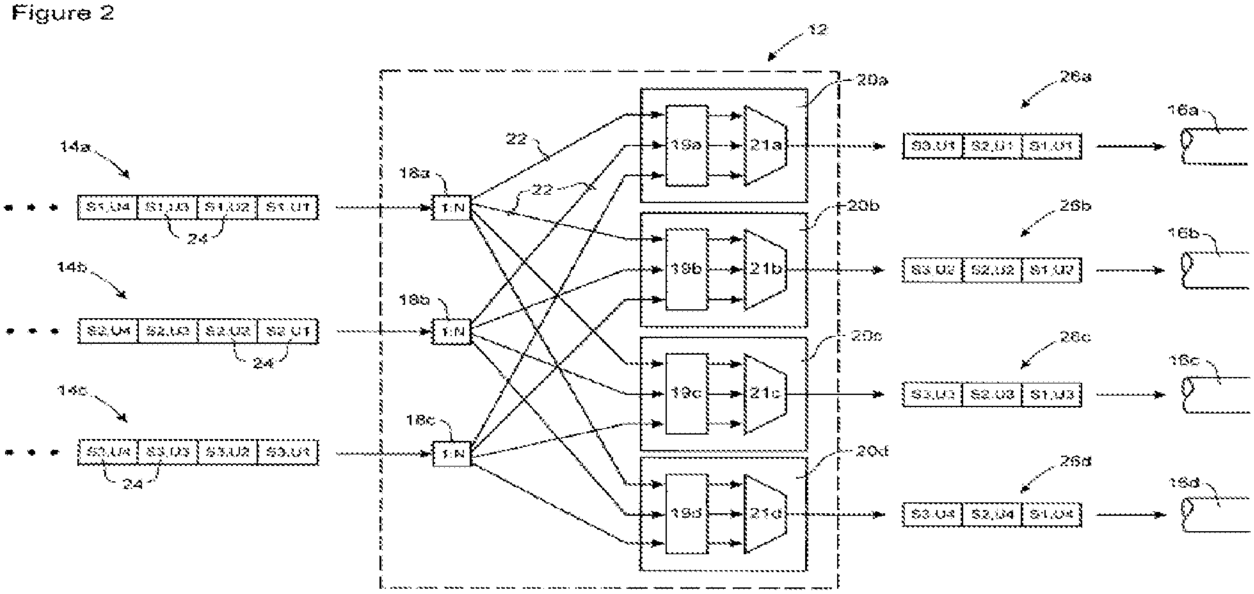
each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."

Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated , which is the source identifier for PDUs is nothing but "Q-IDs."

Referring to claim 13,

Smith teaches a method for receiving information (Fig.3) over a network comprising:

Figure 2



receiving a plurality of sub-streams, wherein said sub-streams are created by decomposing an input datastream of.....a plurality of input datastreams into said sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues, forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data, forming each PDU by selecting the predetermined amount of data from the input datastream, and appending to each PDU a source identifier identifying source of the input datastream~ and each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels, and each of said sub-streams is transported over said network on the corresponding channel, wherein said transporting comprises forming a data frame comprising one or more PDUs and the appended source identifier for each PDU and transmitting the data frame over the corresponding channel~ and a transmission rate of said input datastream is greater than a maximum transmission rate of any one of said channels; and assembling said sub-streams into a reconstructed output datastream. (Col. 5, line 5-col. 7, line 55, col. 5, line 18-27, "The channels 16 can then be routed through conventional signal transmission circuits (not shown) which perform electro/optical conversion and optical multiplexing of the channels 16 into the fiber links 6a e in a manner known in the art. For example, known signal multiplexing methods may be utilized to multiplex the channels 16 onto respective different wavelengths within the same WDM or DWDM fiber, respective wavelengths within two or more WDM or DWDM fibers, or respective single-wavelength fibers.")

<u>Examiner notes that the source identifier for PDUs is nothing but "Q-IDs."</u> Which is illustrated on page 19 of 60/ 270, 444 as follows:

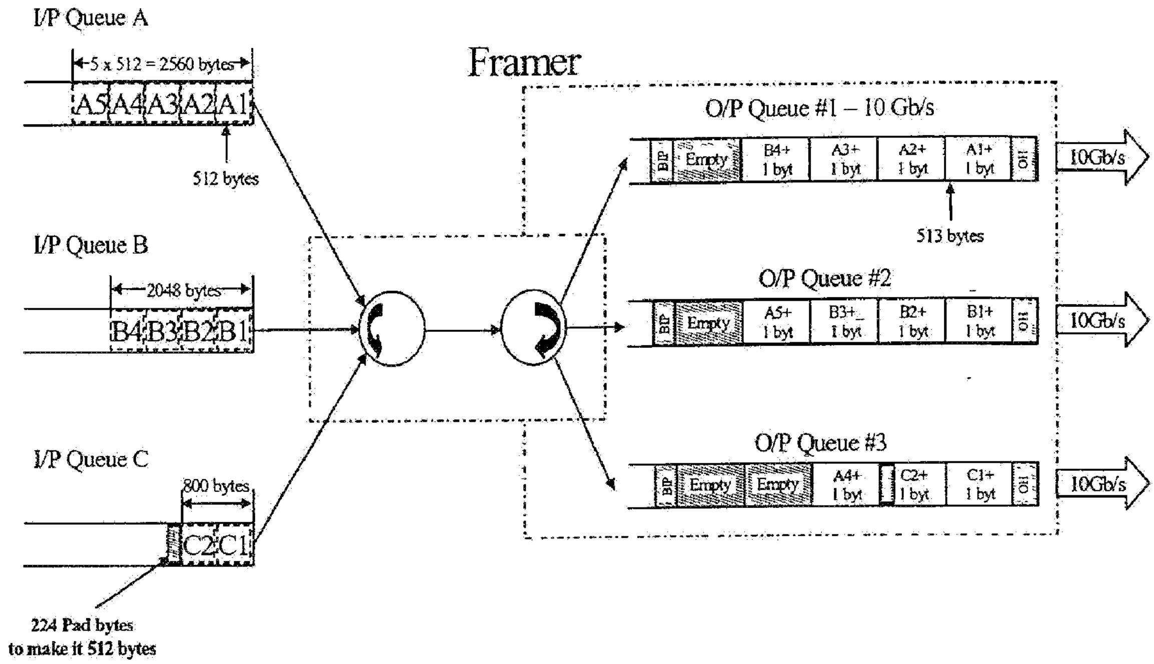


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

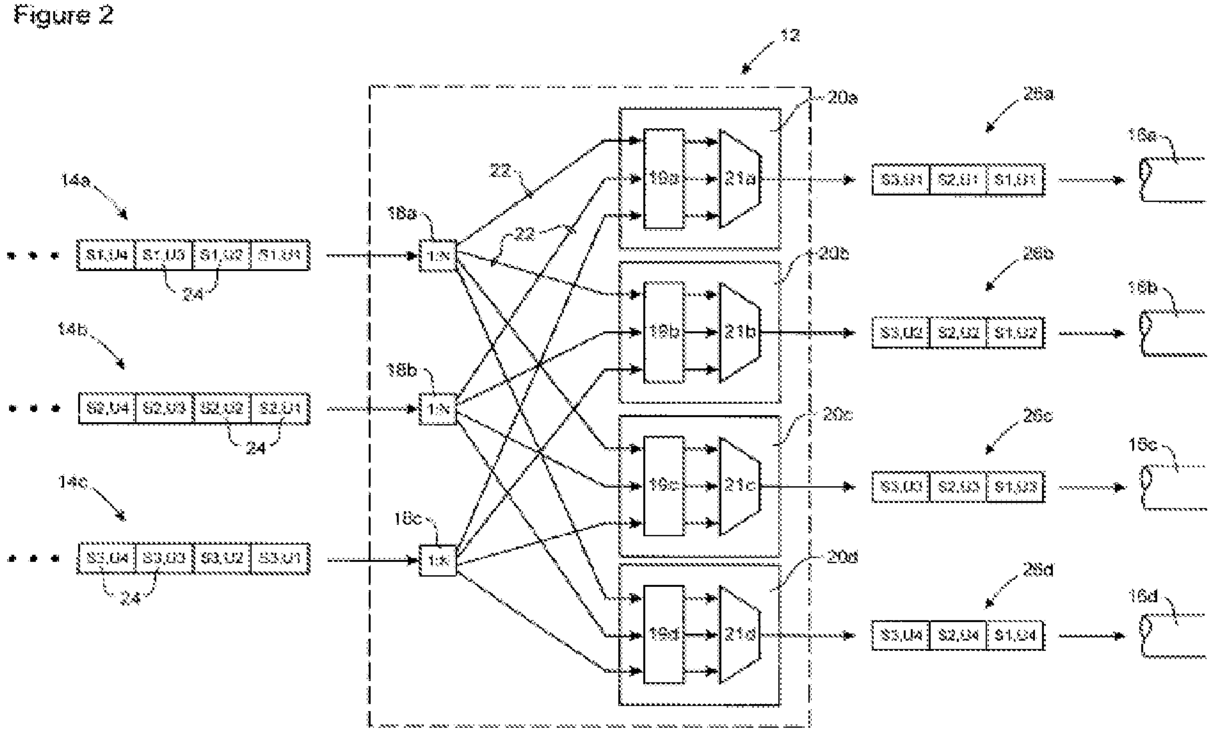
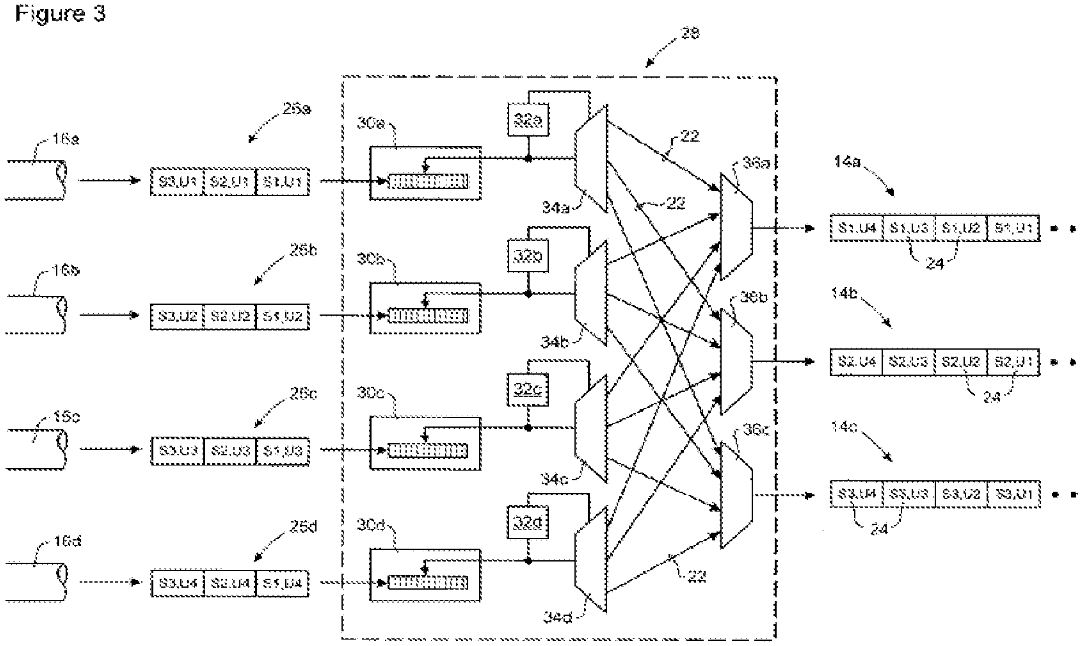


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, “The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. **The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,** and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of

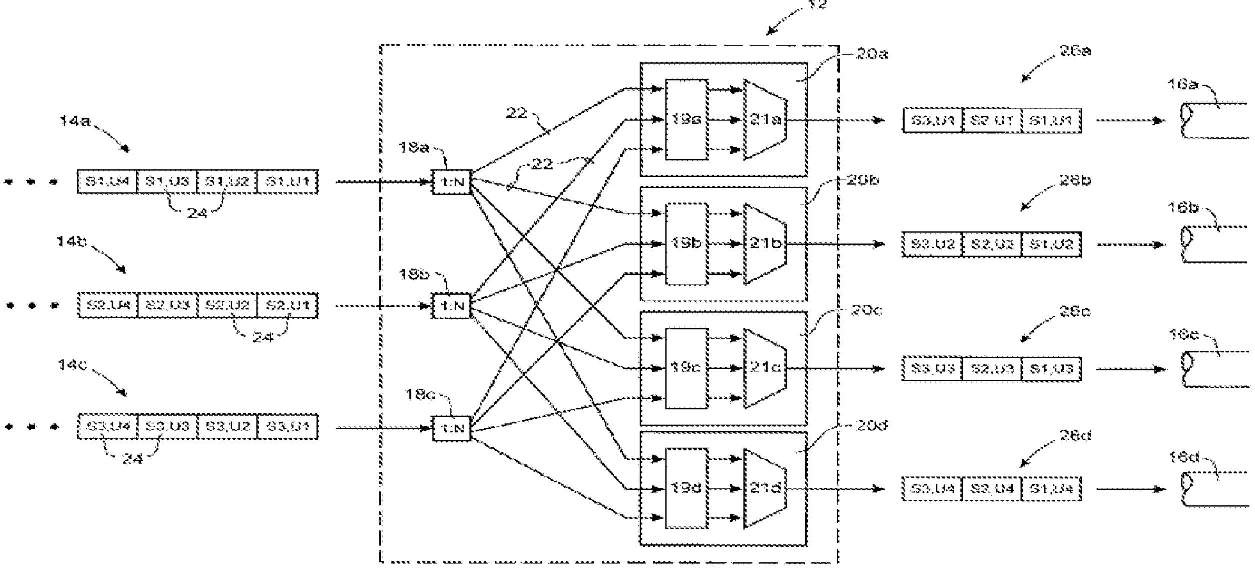
<u>each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."</u>

Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated , which is the source identifier for PDUs is nothing but "Q-IDs."

Referring to claim 24,

Smith teaches an apparatus for transporting information (Fig. 2 and Fig.3) over a network comprising:

Figure 2

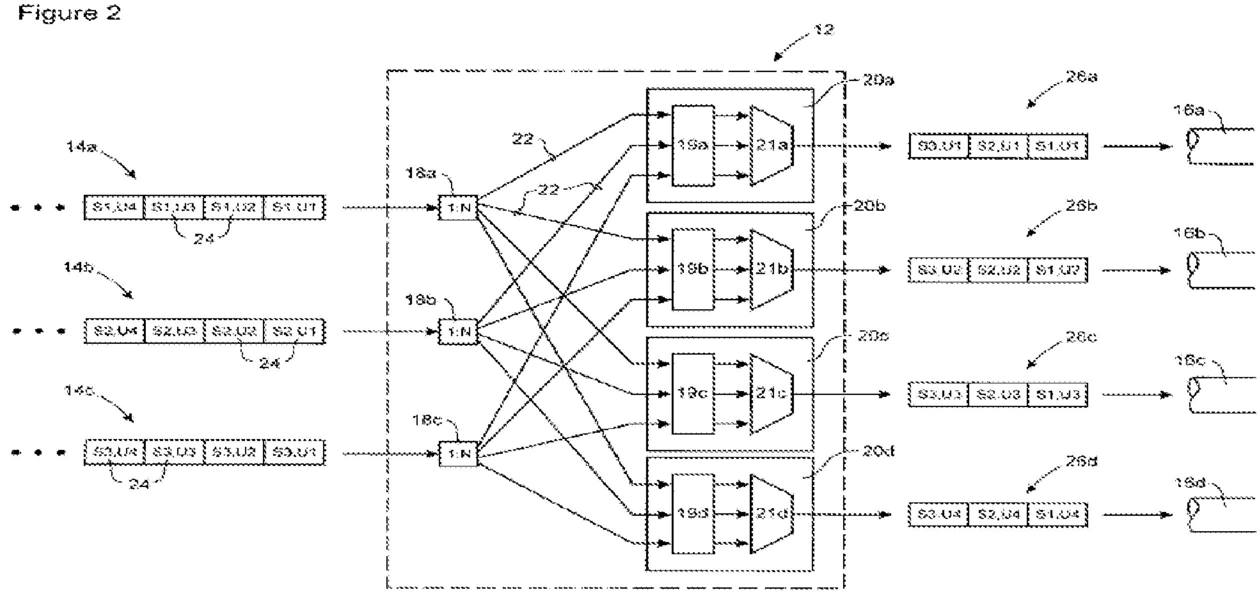


a first sub-stream management device, comprising an input configured to receive an input datastream of a plurality **of input** datastreams, and a plurality of outputs, wherein each of said outputs is configured to output one of a plurality of sub-streams, wherein the input datastream is decomposed to form the plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of the plurality of queues, forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data, forming each PDU by selecting the predetermined amount of data from the input datastream, and appending to each PDU a source identifier identifying the source of the **input** datastream, **and** each of the plurality of queues corresponds to a corresponding channel of a plurality of channels, each of said sub-streams is transported over said network on the corresponding channel, wherein said transporting comprises forming a data frame comprising one or more PDUs and the appended source identifier for each PDU and transmitting he data frame over the corresponding channel, and a transmission rate of said input datastream is greater than a maximum transmission rate of any one of said channels. (Col. 5, line 5-col. 7, line 55, col. 5, line 18-27, "The channels 16 can then be routed through conventional signal transmission circuits (not shown) which perform electro/optical conversion and optical multiplexing of the channels 16 into the fiber links 6a e in a manner known in the art. For example, known signal multiplexing methods may be utilized to multiplex the channels 16 onto respective different wavelengths within the same WDM or DWDM fiber, respective wavelengths within two or more WDM or DWDM fibers, or respective single-wavelength fibers.")

Referring to claim 30,

Smith teaches an apparatus for transporting information (Fig. 2 and Fig.3) over a network comprising:

Figure 2



a first sub-stream management device, comprising an output configured to output a reconstructed output datastream, and a plurality of inputs, wherein each of said inputs is configured to receive one of a plurality of sub-streams, said sub-streams are created by decomposing an input datastream of a plurality of input datastreams into said sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues, forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data, forming each PDU by selecting the predetermined amount of data from the input datastream, and appending to with each PDU a source identifier identifying the source of the input datastream, and each queue of the plurality of queues corresponds to a corresponding channel of a plurality of channels, each of said sub-streams is transported over said network on the corresponding channels~ wherein said transporting comprises forming a data frame comprising one or more PDUs and the appended source identifier for each PDU and transmitting the data frame over the corresponding channel, and a transmission rate of said input datastream is greater than a maximum transmission rate of any one of said channels.

(Col. 5, line 5-col. 7, line 55, col. 5, line 18-27, "The channels 16 can then be routed through conventional signal transmission circuits (not shown) which perform electro/optical conversion and optical multiplexing of the channels 16 into the fiber links 6a e in a manner known in the art. For example, known signal multiplexing methods may be utilized to multiplex the channels 16 onto respective different wavelengths within the same WDM or DWDM fiber, respective wavelengths within two or more WDM or DWDM fibers, or respective single-wavelength fibers.")

<u>Examiner notes that the source identifier for PDUs is nothing but "Q-IDs."</u> Which is illustrated on page 19 of 60/ 270, 444 as follows:

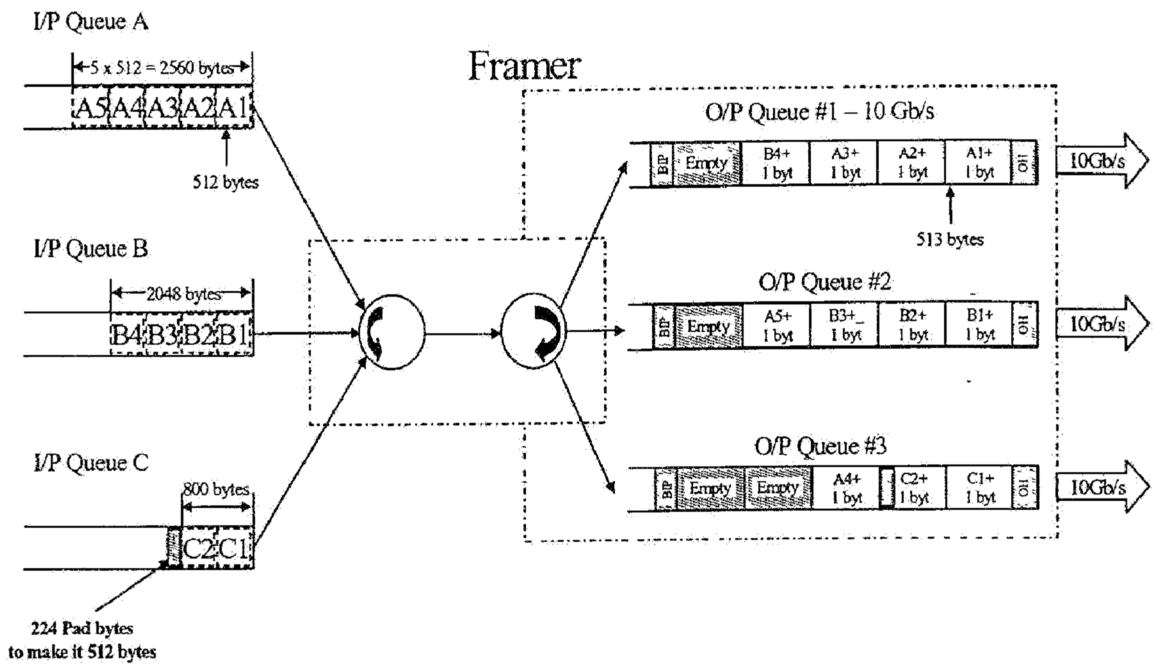


Figure 16 3 input 3 output payload formation example

Smith teaches the following in Figs. 2 and 3:

Figure 2

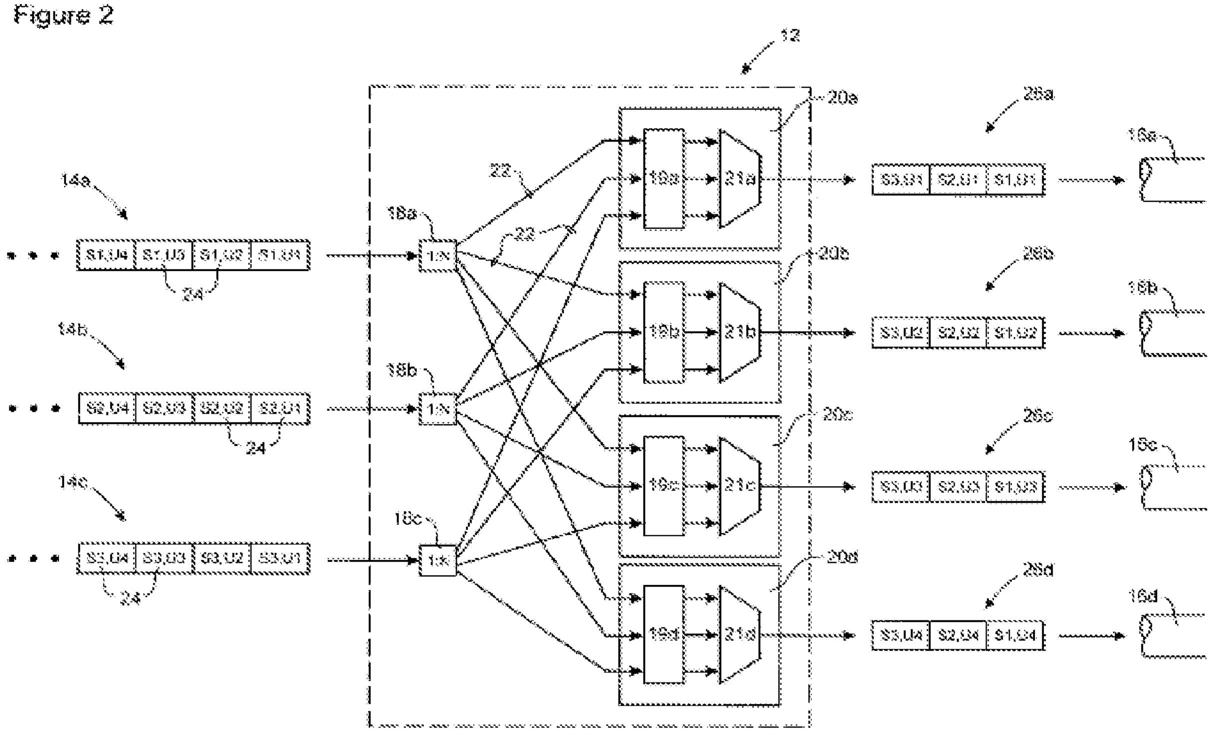
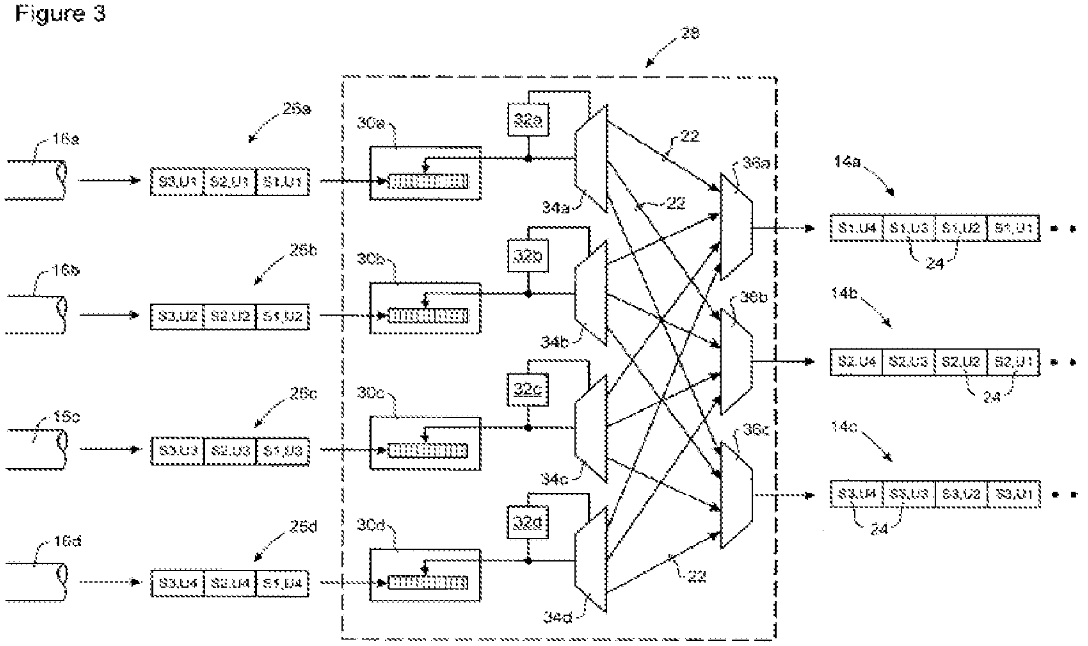


Figure 3



Also Smith teaches at col. 6, line 55-col. 7, line 21, “The signal distribution process described above with respect to FIG. 2 is fully reversible to recover the original data signals 14. FIG. 3 is a block diagram of a signal recovery unit 28, which is implemented in each receiving end node 4a e. The signal recovery unit 28 shown in FIG. 3 can be implemented in hardware and/or software downstream of conventional optical demultiplexing and opto/electrical conversion circuits, which operate to optically demultiplex the channels 16a d from a fiber link 6, and convert each composite data-stream 26a d into electronic form for processing. Conventional clock recovery and signal regeneration circuits (not shown) may also be implemented upstream of the signal recovery circuit 28 shown in FIG. 3.

The signal recovery unit 28 generally includes at least N parallel elastic buffers 30a d, each of which is arranged to receive a respective composite data-stream 26 from one of the channels 16. The elastic buffers 30a d cooperate to de-skew the composite data-streams 26a d, and thus compensate for propagation delay differences between each of the channels. Each of the de-skewed composite data-signals 26a d is then passed to a respective framer 32 and demultiplexor 34. **<u>The framer 32 analyses the respective composite data-stream 26 to detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26,</u>** <u>and generates a synchronization signal which is used to control the operation of the respective demultiplexor 34. Using the synchronization signal, in combination with the known interleaving sequence of the interleavers 20a d, each demultiplexor 30a d operates to demultiplex its respective composite data-stream 26, to recover one sub-stream 22 of</u>

<u>each data signal 14. The sub-streams 22 of each data signal 14 are forwarded (one from each demultiplexor 30) to a one of M multiplexors 32a c which multiplex the sub-streams 22 to recover the respective data signals 14."</u>

<u>Thus as identified in the previous Office Action, Smith teaches to "detect the sub-stream identifiers of each of the sub-streams 22 contained in the composite data-stream 26</u>", as such identified by Smith as shown in Figs 2 and 3 , S1 is an input stream identifier which is a source identifier and U1 is packet identifier which is appended to the PDUs. This is exactly what the Applicant has indicated , <u>which is the source identifier for PDUs is nothing but "Q-IDs."</u>

****

(10) Response to Argument

<u>Appellant's Argument:</u>

I. <u>The rejection of Claims 1-11, 13-22 and 24-37 under 35 U.S.C. §103(a) as being unpatentable over *York* in view of *Smith* is unfounded and should be overturned.</u>

Appellants submit that <u>Smith fails to provide disclosure of the claimed appending a source identifier to each PDU</u>. **(page 8 of Appeal Brief)**

In order to identify the originating datastream for each of the sub-streams that are interleaved, *Smith* purportedly utilizes a sub-stream processor that "operates to insert a unique sub-stream identifier into each of the sub-streams 22 received by the interleaver 20, prior to their being interleaved into a respective channel 16." *Smith* 5:50-54. *Smith* states that "the sub-stream identifier preferably comprises a unique, n-bit word, which is inserted into a respective sub-stream 22 at a predetermined frequency." *Smith* 5:56-59;

see also Smith 6:4-6 ("In general, an insertion frequency of about 8 kHz should yield satisfactory results in most cases."). Thus, *Smith* discloses inserting the disclosed sub-stream identifier into the sub-stream as a separate unit and teaches away from appending a sub-stream identifier to each PDU, as claimed. In fact, *Smith* discloses inserting a substream identifier into the substream without regard for individual PDUs. **(page 9 f Appeal Brief)**

See, Final Office Action, pages 3-7 (emphasis in original). Appellants respectfully submit that neither the cited passage, nor, more broadly, the combination of *Smith and York*, teaches an atomic unit of data, such as a claimed PDU, which contains both a source identifier and payload data. Appellants claim "appending to each PDU a source identifier identifying the source of the input datastream." *Smith* teaches away from the individually source-identifiable atomic data unit claimed by Appellants. Instead, *Smith* purports to teach a source identifier injected at a low frequency interval into a substream of otherwise unidentifiable packets, wherein the atomic unit *(e.g.,* packet) does not carry any source identifier. **Metaphorically, it is as though Appellants claimed a data container (the PDU) with a label indicating its source.** Following the metaphor, *Smith,* if it teaches anything, teaches away from individually source-identified data containers as it purports to teach a line of data containers (packets) without source identifiers on the individual data containers, wherein one of the data containers is used to indicate a source of its anonymous brothers within the stream. **(page 11 of Appeal Brief)**

Examiner's response:

First of all, it must be noted that:

1. Appellant is sheltering to the "metaphor" to explain the claim limitation in question. Examiner would like to point out that **"It is the claims that define the claimed invention, and it is claims, not specifications that are anticipated or unpatentable.** *Constant v. Advanced Micro-Devices Inc.*, 7 USPQ2d 1064.

2. Appellant has completely ignored the teachings of *york.*

3. Arguments regarding claim 1 applies to independent claims 13, 24 and 30 as well.

Following is how Examiner had discerned "a source identifier identifying the source of the input stream" and "the PDU".

Appellant had identified "a source identifier identifying the source of the input stream" being the "Q_IDs" in response dated 01/10/2008 as well as the same is reiterated in this Appeal Brief , page 8.

Based on the "a source identifier identifying the source of the input stream" being "Q_IDs" and the claim language, Examiner had discerned the claim limitations as follows:

Claim 1 recites:

decomposing an input datastream of a plurality of input datastreams into a plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues,

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data,

forming each PDU by selecting the predetermined amount of data from the input datastream.

Thus, each “PDU” is the selected predetermined amount of data from the input datastream which forms “the portion of the input datastream” because claim itself recites that “forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data.”

Thus, “the portion of the input datastream”, which is “PDU”, is then “placed into one of a plurality of queues”, (as claim recites “placing a portion of the input datastream into one of a plurality of queues.)

Thus, “placing a portion of the input datastream into one of a plurality of queues” is “decomposing”.

Thus, if_decomposing is “decomposing an input datastream of a plurality of input datastreams into a plurality of sub-streams wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues”, then “a plurality of queues” are “a plurality of sub-streams.”

And as stated above, “a plurality of queues” which are “a plurality of sub-streams” are “formed” of “the portion of the input datastream”, which is “PDU”. **Therefore each sub-stream is a PDU.**

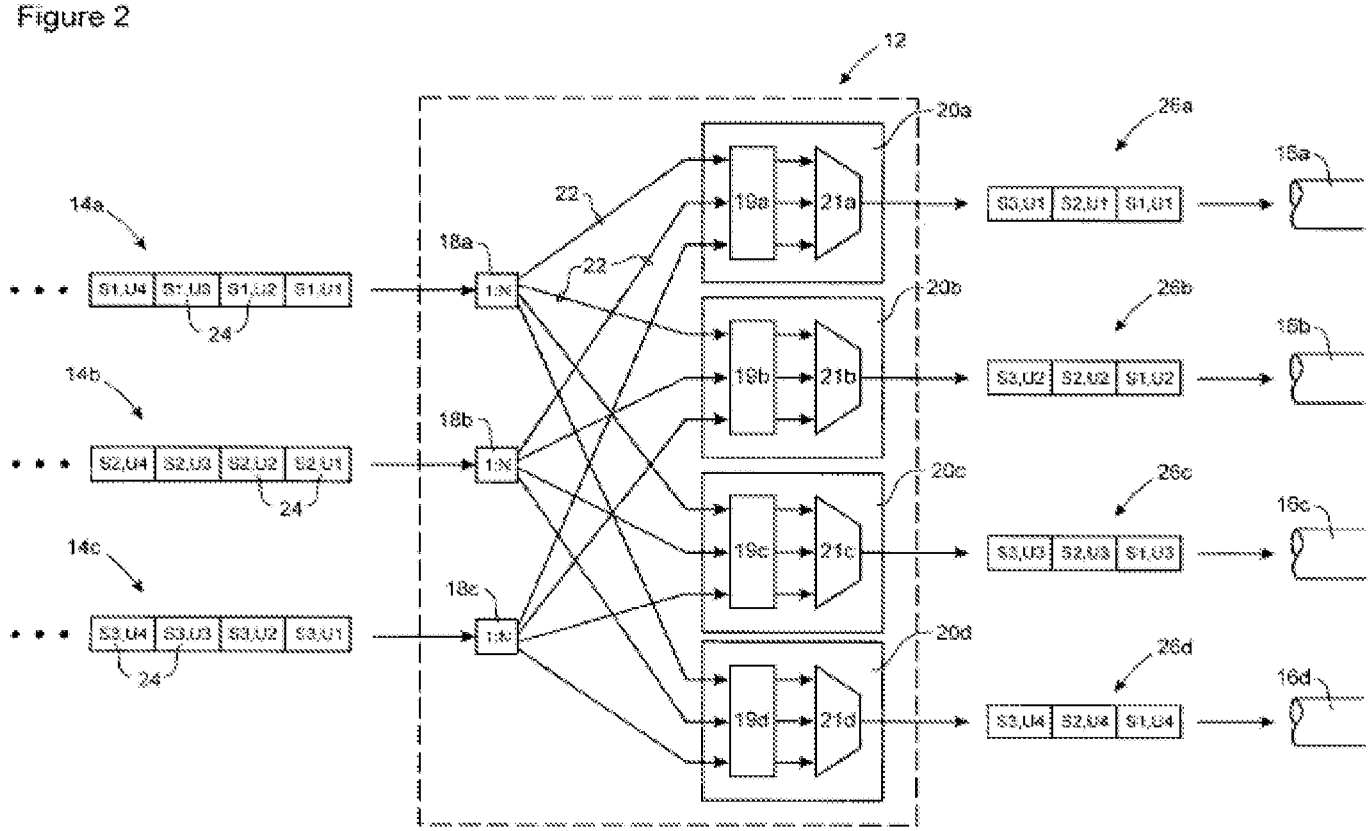
Along with the above explanation and the Appellant’s response stating " a source identifier identifying the source of the input stream” being “Q_IDs”, Examiner had interpreted the claim limitation “appending to each PDU a source identifier identifying the source of the input datastream” as being “appending to each portion of the input

datastream (which is "PDU") a Q_ID (which is "a source identifier) identifying the source of the input datastream."

Based on the above interpretation, Examiner had stated that "York fails to teach "decomposing an input datastream of a plurality of input datastreams, appending to each PDU a source identifier identifying the source of the input datastream".

Smith teaches by depiction at Fig. 2 as follows,

Figure 2



And by description:

At col. 5, line 31-47:

"In general, the signal distribution unit 12 comprises at least M signal dividers 18a c, and at least N interleavers 20a d. Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is <u>to divide each data signal 14 into a sequential series of data units 24 of a predetermined length.</u> The length of each data unit is arbitrary, and may be as short as a single bit. <u>The signal divider 18 then forwards each successive data unit 24 of a respective data signal 14 to a respective one of the sub-streams 22, in turn, so that each sub-stream 22 includes a substantially equal proportion of the original data signal 14.</u> Each sub-stream 22 from a data signal 14 is forwarded to a respective one of the interleavers 20a b, so that each channel 16 carries one sub-stream 22 from (and thus a substantially equal portion of) each data signal 14."

At col. 3, line 30-31, "<u>Each data signal</u> may be a Forward Error Correction (FEC) <u>encoded data stream</u>.

Please note that Smith has datastreams (each data signal 14a-14c) consist of "<u>a sequential series of data units 24 of a predetermined length</u> and each of these "input stream" is decomposed into four "substreams 22 by "The signal divider 18" which "then forwards <u>each successive data unit 24 of a respective data signal 14 to a respective one of the sub-streams 22</u>, in turn, so that each sub-stream 22 includes a substantially equal proportion of the original data signal 14.

Thus, Smith teaches decomposing "data signal" which is " an input stream" (Fig. 2, element (14a) of a plurality of input datastreams (14a, 14b, 14c) into a plurality of

sub-streams (sub-streams are Fig. 2, elements 22) wherein each substream 22 is "each successive data unit 24 ("the portion of the input datastream", which is "PDU") of a respective data signal 14" and thus is "a plurality of queues" ("a plurality of sub-streams") are "formed" of "the portion of the input datastream", which is "PDU".)

Smith further teaches at col. 5, line 48-56, "Each interleaver 20, which may be implemented in hardware and/or software, comprises a sub-stream processor 19 and a sequential interleaving multiplexer 21. The sub-stream processor operates to insert a unique sub-stream identifier into each of the sub-streams 22 received by the interleaver 20, prior to their being interleaved into a respective channel 16. The sub-stream identifier serves to enable successful discrimination and separation of the sub-streams from a respective channel 16 at a downstream node 4."

This "inserting unique sub-stream identifier into each of the sub-streams (a Q_ID) is "appending to each PDU a source identifier <u>identifying</u> the source of the input datastream" as claim states that <u>"a plurality of queues" are "a plurality of sub-streams" which are "formed" of</u> "the portion of the input datastream" wherein "the portion of the input datastream" is "PDU". Please note that each substream 22 is "each successive data unit 24 ("the portion of the input datastream", which is "PDU") of a respective "data signal 14" as depicted in Fig, 2.

Thus Smith teaches "appending to each PDU a source identifier identifying the source of the input datastream."

<u>Appellant's Argument:</u>

In addition, Appellants submit that a person of ordinary skill in the art would not be motivated to combine *York with Smith* because such a combination would not be

successful. *York* is designed to purportedly transport one input datastream over a plurality of transmission links. In order to do so, *York* relies upon a preset ordering of transmitter queues that is transmitted to the receiver in order to facilitate reassembling the datastream. No identification of the transmitted packets is made or necessary. Instead, he reassembly is performed on a queue by queue basis (or transmit line by transmit line basis). *See York* 4:6-17, 4:35-41, 6:19-24. *York* does not contemplate multiple input datastreams, nor can *York* handle multiple input datastreams. In fact, were one to put multiple input datastreams into the *York* device, the output at the receiving end would be meaningless jumble because *York* provides no mechanism for identifying input datastreams. *Smith* cannot remedy this deficiency, because of the differing purpose for which *Smith* is designed. **(pages 11 and 12 of Appeal Brief)**

Further, one would not be motivated to combine *Smith* with *York* because *York* does not provide disclosure that would enhance the performance of *Smith. Smith* provides a mechanism for purportedly dealing with multiple input datastreams, splitting the datastreams, transmitting a number of transmission datastreams, and then reassembling the original datastreams from the transmission datastreams. Thus, combining the teachings of *York* with *Smith* would be at best redundant. In addition, *Smith* provides transmission of the transmission datastreams using a serial-composite datastream. The Final Office Action has failed to provide a basis for any benefit that one would derive from coupling the packetized system of *York* to *Smith.* **(page 12 of Appeal Brief)**

See, Final Office Action, pages 7-8. The Final Office Action states that "it would have been obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose." Applicants respectfully submit that "broad conclusive statements about the teaching of multiple references, standing alone, are not 'evidence.'" *See Ruiz v. A.B.*

Chance Co., 234 F.3d 654, 666 (Fed. Cir. 2000). The Final Office action's argument essentially requires belief, without evidence or supporting logic, that an amalgam of unrelated signaling systems would result in a coherent signal. Appellants respectfully submit that the art does not teach this "predictability" to which the Final Office Action alludes but does not cite. Instead, it teaches two different systems designed to achieve different purposes using different methods, and the Final Office Action offers no teaching to bridge the two. **(page 13 of Appeal Brief)**

Examiner's response:

"it would have obvious to one skilled in the art to substitute one method for the other to achieve the predictable results of decomposing the input datastream when it is just one to decompose or more than one to decompose." is a conclusive statement that was made based on the teachings of York and Smith. Because York and Smith teaches decomposing input datastreams, one incorporating the invention for "decomposing an input datastream" and the other incorporating the invention for "decomposing the plurality of input datastreams" respectively. The difference of the invention lays into the inventive component they use, "Data Splitter" of York and "The signal distributor unit "of Smith.

York sends data packets in round-robin fashion after splitting just an input datastream and uses the same technique to reassemble the data stream at the receiver end. There is no need to identify the packet as there is only one data stream.

Smith send data units from multiple input stream by splitting multiple input stream and therefore identification of the data units are necessary to reassemble the each of

the multiple data streams at the receiver end. This function is performed by the signal distributor unit 12 of Fig. 2.

Appellant's Argument:

II. Rejection of Claims 1, 13, 24 and 30 under 35 U.S.C. §102(e) as being anticipated by Smith is unfounded and should be overturned.

Smith does not disclose appending a sub-stream identifier to a PDU, as claimed. In fact, *Smith* discloses inserting a substream identifier into the substream without regard for individual PDUs **(page 15 of Appeal Brief)**

But *Smith* is clear that the *Smith's* sub-stream identifier is inserted into the sub-stream created by these sequential series of data units and is not appended to each of *Smith's* data units. *See Smith* 5:50-6:6. In addition, *Smith* fails to provide any disclosure of forming a "data frame comprising one or more PDUs and the appended source identifier for each PDU and transmitting the data frame over the corresponding channel," as claimed. Without such disclosure, *Smith* cannot be said to anticipate the claims. **(page 15 of Appeal Brief)**

Appellants claimed a data container (the PDU) with a label indicating its source. *Smith,* if it teaches anything, teaches away from individually source-identified data containers as it purports to teach a line of data containers (packets) without source identifiers on the individual data containers, wherein one of the data containers is used to indicate a source of its anonymous brothers within the stream. **(page 17 of Appeal Brief)**

Examiner's response:

First of all, it must be noted that:

1. Appellant is stating "Appellants claimed a data container (the PDU) with a label indicating its source." to explain the claim limitation in question. Examiner would like to point out that **"It is the claims that define the claimed invention, and it is claims, not specifications that are anticipated or unpatentable.** *Constant v. Advanced Micro-Devices Inc.*, 7 USPQ2d 1064.

2. Arguments regarding claim 1 applies to independent claims 13, 24 and 30 as well.

Following is how Examiner had discerned “a source identifier identifying the source of the input stream” and “the PDU”.

Appellant had identified “a source identifier identifying the source of the input stream” being the “Q_IDs” in response dated 01/10/2008 as well as the same is reiterated in this Appeal Brief , page 8.

Based on the “a source identifier identifying the source of the input stream” being “Q_IDs” and the claim language, Examiner had discerned the claim limitations as follows:

Claim 1 recites:

decomposing an input datastream of a plurality of input datastreams into a plurality of sub-streams, wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues,

forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data,

forming each PDU by selecting the predetermined amount of data from the input datastream.

Thus, each “PDU” is the selected predetermined amount of data from the input datastream which forms “the portion of the input datastream” because claim itself recites that “forming the portion of the input datastream using one or more payload data units (PDUs) each comprising a predetermined amount of data.”

Thus, “the portion of the input datastream”, which is “PDU”, is then “placed into one of a plurality of queues”, (as claim recites “placing a portion of the input datastream into one of a plurality of queues.)

Thus, “placing a portion of the input datastream into one of a plurality of queues” is “decomposing”.

Thus, if decomposing is “decomposing an input datastream of a plurality of input datastreams into a plurality of sub-streams wherein said decomposing comprises placing a portion of the input datastream into one of a plurality of queues”, <u>then “a plurality of queues” are “a plurality of sub-streams.”</u>

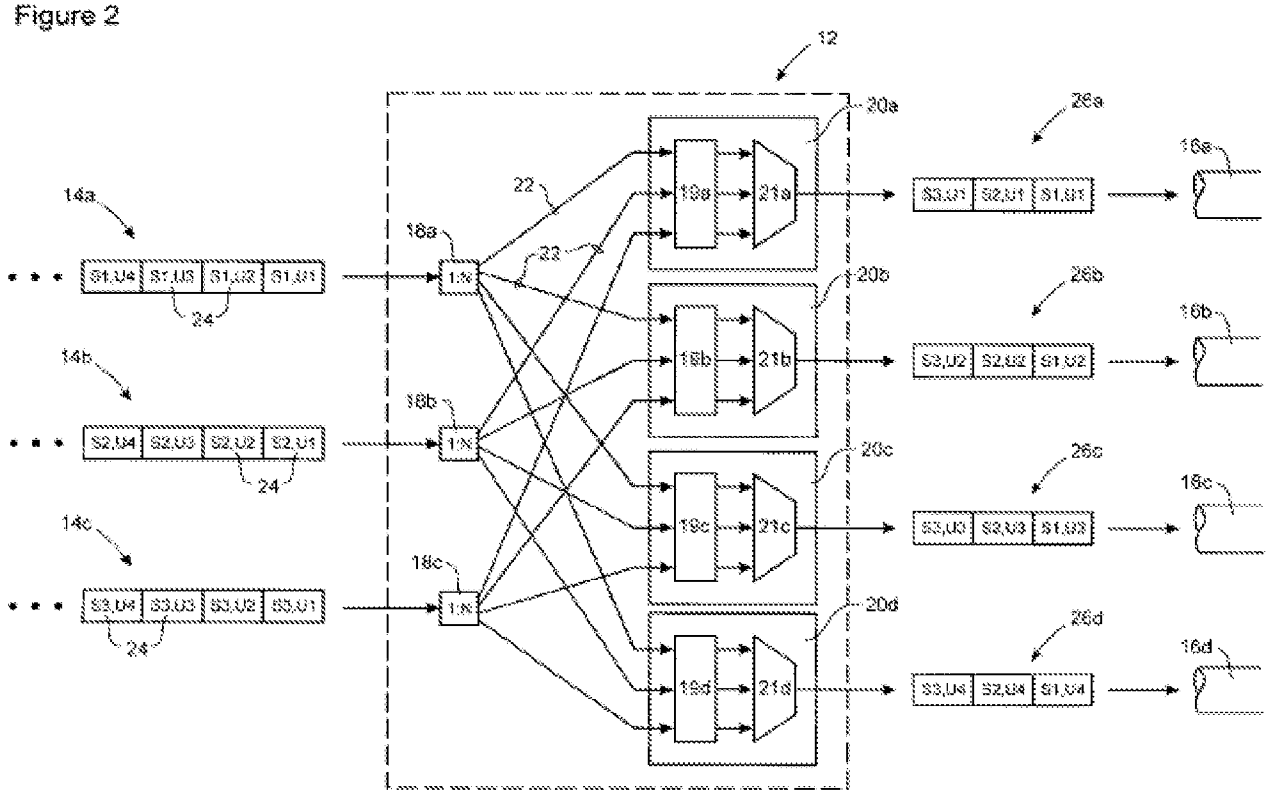
<u>And as stated above, “a plurality of queues” which are “a plurality of sub-streams” are “formed” of</u> “the portion of the input datastream”, which is “PDU”. <u>**Therefore each sub-stream is a PDU.**</u>

Along with the above explanation and the Appellant’s response stating " a source identifier identifying the source of the input stream” being “Q_IDs”, Examiner had interpreted the claim limitation “appending to each PDU a source identifier identifying the source of the input datastream” as being “appending to each portion of the input datastream (which is “PDU”) a Q_ID (which is “a source identifier) <u>identifying</u> the source of the input datastream.”

Based on the above interpretation, Examiner had stated that “York fails to teach “decomposing an input datastream <u>of a plurality of input datastreams,</u> appending to each PDU a source identifier identifying the source of the input datastream”.

Smith teaches by depiction at Fig. 2 as follows,

Figure 2



And by description:

At col. 5, line 31-47:

"In general, the signal distribution unit 12 comprises at least M signal dividers 18a c, and at least N interleavers 20a d. Each signal divider 18a c divides a respective data signal 14a c into N sub-streams 22 (i.e. one sub-stream 22 for each channel 16). As illustrated in FIG. 2, one method by which this may be accomplished is to divide each data signal 14 into a sequential series of data units 24 of a predetermined length. The length of each data unit is arbitrary, and may be as short as a single bit. The signal divider 18 then forwards each successive data unit 24 of a respective data signal

14 to a respective one of the sub-streams 22, in turn, so that each sub-stream 22 includes a substantially equal proportion of the original data signal 14. Each sub-stream 22 from a data signal 14 is forwarded to a respective one of the interleavers 20a b, so that each channel 16 carries one sub-stream 22 from (and thus a substantially equal portion of) each data signal 14."

At col. 3, line 30-31, "Each data signal may be a Forward Error Correction (FEC) encoded data stream.

Please note that Smith has datastreams (each data signal 14a-14c) consist of "a sequential series of data units 24 of a predetermined length and each of these "input stream" is decomposed into four "substreams 22 by "The signal divider 18" which "then forwards each successive data unit 24 of a respective data signal 14 to a respective one of the sub-streams 22, in turn, so that each sub-stream 22 includes a substantially equal proportion of the original data signal 14.

Thus, Smith teaches decomposing "data signal" which is " an input stream" (Fig. 2, element (14a) of a plurality of input datastreams (14a, 14b, 14c) into a plurality of sub-streams (sub-streams are Fig. 2, elements 22) wherein each substream 22 is "each successive data unit 24 ("the portion of the input datastream", which is "PDU") of a respective data signal 14" and thus is "a plurality of queues" ("a plurality of sub-streams") are "formed" of "the portion of the input datastream", which is "PDU".)

Smith further teaches at col. 5, line 48-56, "Each interleaver 20, which may be implemented in hardware and/or software, comprises a sub-stream processor 19 and a sequential interleaving multiplexer 21. The sub-stream processor operates to insert a

unique sub-stream identifier into each of the sub-streams 22 received by the interleaver 20, prior to their being interleaved into a respective channel 16. The sub-stream identifier serves to enable successful discrimination and separation of the sub-streams from a respective channel 16 at a downstream node 4."

This "inserting unique sub-stream identifier into each of the sub-streams (a Q_ID) is "appending to each PDU a source identifier identifying the source of the input datastream" as claim states that "a plurality of queues" are "a plurality of sub-streams" which are "formed" of "the portion of the input datastream" wherein "the portion of the input datastream" is "PDU". Please note that each substream 22 is "each successive data unit 24 ("the portion of the input datastream", which is "PDU") of a respective "data signal 14" as depicted in Fig, 2.

Thus Smith teaches "appending to each PDU a source identifier identifying the source of the input datastream."

(11) Related Proceeding(s) Appendix

For the above reasons, it is believed that the rejections should be sustained.

(Note: the Examiner has made an earnest effort to properly address each and every Appellant's arguments of the appeal brief. In any event or reason if more explanation is needed, the Examiner will gladly provide as necessary).

Respectfully submitted,

/Ashok B. Patel/

Examiner, Art Unit 2456

Conferees:

/Bunjob Jaroenchonwanit/

Supervisory Patent Examiner, Art Unit 2456

/Jeffrey Pwu/

Supervisory Patent Examiner, Art Unit 2446